

Penghargaan

Bismillahirrahmanirrahim

Universiti Malaya

Perpustakaan SKTM

Sistem Pengkategorian Teks

Oleh

Hazyzyizuan Abu Hassan
WEK000003

Dibawah penyelia
Cik Norisma Idris

Dan Moderator
Encik Mohd Noridzuan

Dihantar kepada
Fakulti Sains Komputer Dan Teknologi Maklumat
Universiti Malaya

Untuk memenuhi keperluan bagi
Ijazah Pertama
Sains Komputer
Sesi 2002/2003

Penghargaan

Bismillahirrahmanirahim

Pujian-pujian dan syukur ke hadrat Yang Maha Esa kerana dengan izinnya maka latihan ilmiah 1 ini dapat disiapkan pada masa yang telah ditetapkan.

Sekalung budi dan terima kasih kepada individu-individu yang terlibat dalam usaha untuk menyiapkan projek ini. Tanpa bantuan mereka dengan izin Allah mungkin projek ini tidak dapat disiapkan seperti yang dijangka.

Penghargaan dan penghormatan kepada penyelia, Cik Norisma Idris yang ikhlas membantu dan memberikan garis panduan dan nasihat disepanjang penglibatan saya dalam projek ini.

Setulus ingatan kepada ibu dan ayah yang banyak membantu dari segi kewangan dan sokongan serta iringan doa yang berpanjangan.

Terima Kasih kepada semua yang terlibat dan jasa kalian hanya Allah dapat membalasnya.

Abstrak

Projek Latihan Ilmiah merupakan salah satu syarat wajib untuk menamatkan Kursus Ijazah Pertama Sains Komputer Universiti Malaya dengan kepujian.

Sistem Pengkategorian Teks merupakan aplikasi yang menyediakan kemudahan kepada pengguna untuk melaksanakan tugas mengkategorikan dokumen mengikut kategori-kategori tertentu.

Dalam konteks pembangunan, kriteria-kriteria yang penting dipertimbangkan dari sudut pemprosesan bahasa semulajadi (Natural Language Processing) yang menjadi panduan untuk tujuan pembangunan terutamanya dari aspek rekabentuk aplikasi. Teknik dalam kepintaran buatan iaitu Nearest Neighbour telah digunakan didalam pembangunan Sistem Pengkategorian Teks. Perisian pembangunan Sistem Pengkategorian Teks menggunakan Visual C++ . Sistem pengendalian menggunakan Windows 98 .

Di dalam laporan ini juga akan menerangkan penyediaan fasa pembangunan dan implementasi, penulisan aturcara yang lengkap yang disediakan oleh AppWizwrd MFC di dalam Visual C++ bagi memudahkan pembangun sistem membangunkan aturcara dan pengujian di mana sistem akan diuji dengan data sebenar. Pada bab penyediaan dokumentasi akan menerangkan bagaimana dokumentasi dilakukan dan akan menerangkan penyelenggaraan serta cadangan pembangunan masa hadapan.

Isi Kandungan

Penghargaan	i
Abstrak	ii
Isi Kandungan	iii
Senarai Gambarajah	viii
Senarai Jadual	x
Bab 1 : Pengenalan	1
1.1 Definasi Projek	1
1.2 Pendekatan Yang Digunakan	2
1.3 Objektif Projek	2
1.4 Kategori-Kategori Utama	3
1.5 Kelebihan	3
1.6 Sasaran Pengguna	3
1.7 Jangka Masa Projek	4
1.8 Mengenalpati Masalah	5
1.9 Struktur Laporan	6

Bab 2	: Kajian Literasi	8
2.1	Pengenalan Kepada Kajian Literasi	8
2.2	Pengenalan Kepada Pengkategorian Teks	9
2.3	Contoh Berita Yang Belum Diproses	11
2.4	Pengenalan Kepada Pemprosesan Bahasa Tabii	13
2.5	Algoritma Nearest Neighbour	16
2.5.1	Algoritma K-Nearest Neighbor	16
2.5.2	Matrik Jarak	19
2.6	Pemprosesan Indeks	20
2.6.1	Hapuskan Perenggan	21
2.6.2	Hapuskan titik,koma,hypen	22
2.6.3	Saperator Indicator	22
2.6.4	Hapuskan Nombor Dan Simbol Istimewa	23
2.6.5	Menukar Huruf Besar Ke Huruf Kecil	23
2.6.6	Hapus Stopword	24
2.6.7	Word Stemming	26
2.6.8	Pengekstrakan Maklumat	26
2.7	Analisa Sistem Yang Sedia Ada	28
2.7.1	Pengenalan Kepada Sistem Pengkategorian Teks Yang Ada	28
2.7.2	Textcat	28

2.8	Perisian Pembangunan Aplikasi	30
2.8.1	Visual C++	31
Bab 3	: Metadologi	32
3.1	Pengenalan	32
3.2	Pertimbangan Dan Analisis Metodologi	34
3.2.1	Model Air Terjun	34
3.2.1.1	Kelebihan dan Kekurangan Model Air Terjun	36
3.2.2	Model Lingkaran(Spiral)	37
3.2.3	Model Air Terjun dengan Prototaip	38
3.2.3.1	Langkah-Langkah Pemprototaipan	41
3.2.3.2	Kebaikan Dan Keburukan Prototaip	42
3.3	Analisis Dan Sintesis	43
3.4	Fokus Terperinci Setiap Fasa Berdasarkan Model Air Terjun Dengan Prototaip	45
Bab 4	: Analisis Dan Rekabentuk Sistem	50
4.1	Pengenalan	50
4.2	Spesifikasi Keperluan Sistem	50
4.2.1	Keperluan Fungsian	51
4.2.2	Keperluan Bukan Fungsian	51
4.3	Rekabentuk Modul	54

4.3.1	Modul Pengekstrakan Maklumat	56
4.3.2	Modul Perbandingan	57
4.3.3	Modul Antaramuka	58
4.4	Rekabentuk Sistem	59
4.4.1	Senibina Sistem	59
4.4.1.1	Pemprosesan Indeks	60
4.4.1.2	Fungsi Jarak	61
4.4.1.3	Keyword	62
4.5	Rekabentuk Antaramuka	67
Bab 5	: IMPLEMENTASI	70
5.1	Pembangunan dan Implementasi	70
5.2	Objektif fasa pembangunan dan implementasi	71
5.2.1	Fasa pembangunan	72
5.3	Awal pembangunan	74
5.4	Penulisan aturcara dan pengujian	78
5.4.1	Algoritma Program Pengkategorian Teks	78
5.4.1.1	Algoritma untuk dialog memasukkan teks	78

5.4.1.2	Algoritma untuk dialog memasukkan keyword dalam pengkalan data	80
5.4.2	Pengujian Modul Dan Unit	81
5.4.2.1	Pengujian sistem	83
5.4.3	Penyediaan Dokumentasi	85
5.4.3.1	Dokumentasi Teknikal	85
5.4.3.2.1	Dokumentasi Aturcara WordCategorizationDlg.cpp	88
5.4.3.2.2	Dokemntasi Aturcara KeywordDlg.cpp	99
5.4.3.4	Dokumentasi Pengguna	102
5.5	Kelemahan Dan Kebaikan	105
BAB 7	: PENYELENGGARAAN	106
BAB 8	: PEMBANGUNAN MASA HADAPAN	107
Bibliografi		109

Gambarajah

Rajah 1.1	Kategori-kategori dalam berita	3
Rajah 2.1	Contoh berita	11
Rajah 2.2	Contoh berita	11
Rajah 2.3	Contoh berita	12
Rajah 2.4	k -NN Regression	18
Rajah 2.5	Hapuskan Perenggan	21
Rajah 2.6	Hapuskan titik,koma,hypen	22
Rajah 2.7	Saperator Indicator	22
Rajah 2.8	Hapuskan Nombor Dan Simbol Istimewa	23
Rajah 2.9	Menukar Huruf Besar Ke Huruf Kecil	23
Rajah 2.10	Rajah Antaramuka TextCat	29
Rajah 3.1	Model Air Terjun	36
Rajah 3.2	Model Air Terjun dengan Prototaip	41
Rajah 3.3	Perlaksanaan Prototaip	43
Rajah 4.1	Carta Alir Sistem	57
Rajah 4.2	Modul Pengekstrakan Maklumat	58
Rajah 4.3	Modul Perbandingan	59
Rajah 4.4	Modul Antaramuka	60
Rajah 4.5	Senibina Pengkategorian Teks	61
Rajah 4.6	Senibina Pemprosesan Indeks	62
Rajah 4.7	Perlaksanaan Matrik Jarak	63

Rajah 4.8	Pengantaramuka sistem yang dicadangkan	66
Rajah 5.1	Buka File	74
Rajah 5.2	MFC Langkah 1	75
Rajah 5.3	MFC Langkah 2	75
Rajah 5.4	MFC Langkah 3	76
Rajah 5.5	MFC Langkah 4	76
Rajah 5.6	AppWizard Menghasilkan Rangka Projek	77
Rajah 5.7	Rajah masej untuk memeriksa fungsi	82
Rajah 5.8	Dialog About	86
Rajah 5.9	Dialog Keyword Data Entry	86
Rajah 5.10	Dialog Text Categorization	87
Rajah 5.11	Dialog Text Categorization	102
Rajah 5.12	Dialog Keyword Data Entry	103
Rajah 5.13	Dialog Amaran	104

BAB 1 : PENGENALAN

1.1 Definisi Projek

Jadual

Sistem ini dibangunkan untuk mengkategorikan suatu artikel atau dokumen ke

Jadual 1.1 Carta Gant Projek 4

Jadual 2.1 Senarai Stopword 25

dalam artikel. Secara umumnya pengkategorian teks bermaksud mengkategorikan teks bahasa semulajadi secara automatik ke kategori-kategori tertentu berdasarkan kepada kandungan dokumen. Ia adalah sistem termasuk mengindeks teks yang menyokong mengeluarkan dokumen, mengeluarkan data dari sistem aplikasi primari sistem pengkategorian teks yang menentukan kategori objek kepada dokumen untuk menyokong pengambilan maklumat (information retrieval), atau membantu seseorang pengindeks menentukan kategori tertentu. Komponen pengkategorian teks dilihat meningkatkan lagi penggunaan sistem pemprosesan bahasa semulajadi untuk mengeluarkan data. Pengkategorian teks ini juga dilihat sebagai satu komponen penapis untuk sesuatu bidang.

Sistem pengkategorian teks cuba untuk menghasilkan semula penilaian pengkategorian manusia. Satu pendekatan untuk membina sistem pengkategorian teks ini ialah secara manual menempatkan sesetengah set dokumen ke kategori dan kemudian menggunakan pembelajaran induktif untuk menentukan secara automatik kategori berdasarkan kandungan perkataan.

BAB 1 : PENGENALAN

1.1 Definasi Projek

Sistem ini dibangunkan untuk mengkategorikan suatu artikel atau dokumen ke kategori yang tertentu seperti sukan,hiburan,perniagaan dan lain-lain.Beberapa teknik dalam kepintaran buatan iaitu Nearest Neighbour digunakan untuk menklasifikasikan artikel .Secara umumnya pengkategorian teks bermaksud mengkategorikan teks bahasa semulajadi secara automatik ke kategori-kategori tertentu berdasarkan kepada kandungan dokumen. Ia adalah sistem termasuk mengindekkan teks yang menyokong mengeluarkan dokumen, mengeluarkan data dari teks.Satu aplikasi primari sistem pengkategorian teks yang menentukan kategori subjek kepada dokumen untuk menyokong pengambilan matlumat (information retrieval), atau membantu seseorang pengindeks menentukan kategori tertentu. Komponen pengkategorian teks dilihat meningkatkan lagi penggunaan sistem pemprosesan bahasa semulajadi untuk mengeluarkan data . Pengkategorian teks ini juga dilihat sebagai satu komponen penapis untuk sesetengah bidang.

Sistem pengkategorian teks cuba untuk menghasilkan semula penilaian pengkategorian manusia.Satu pendekatan untuk membina sistem pengkategorian teks ini ialah secara manual menempatkan sesetengah set dokumen ke kategori,dan kemudian menggunakan pembelajaran induktif untuk menentukan secara automatik kategori berdasarkan kandungan perkataan .

1.4 Kategori-kategori Utama

1.2 Pendekatan yang digunakan

Terdapat banyak pendekatan-pendekatan yang digunakan untuk mengkategorikan teks iaitu Machine Learning, Statistical Classification, Bayesian Networks, Information Retrieval, Case-Based Reasoning, Language Modeling, Nearest Neighbour. Dalam projek ini pendekatan yang digunakan ialah Pembelajaran Mesin (Machine Learning) yang mana ia menggunakan pengelasan Nearest Neighbour untuk mengkategorikan teks. Penerangan lanjut pada bab 2.

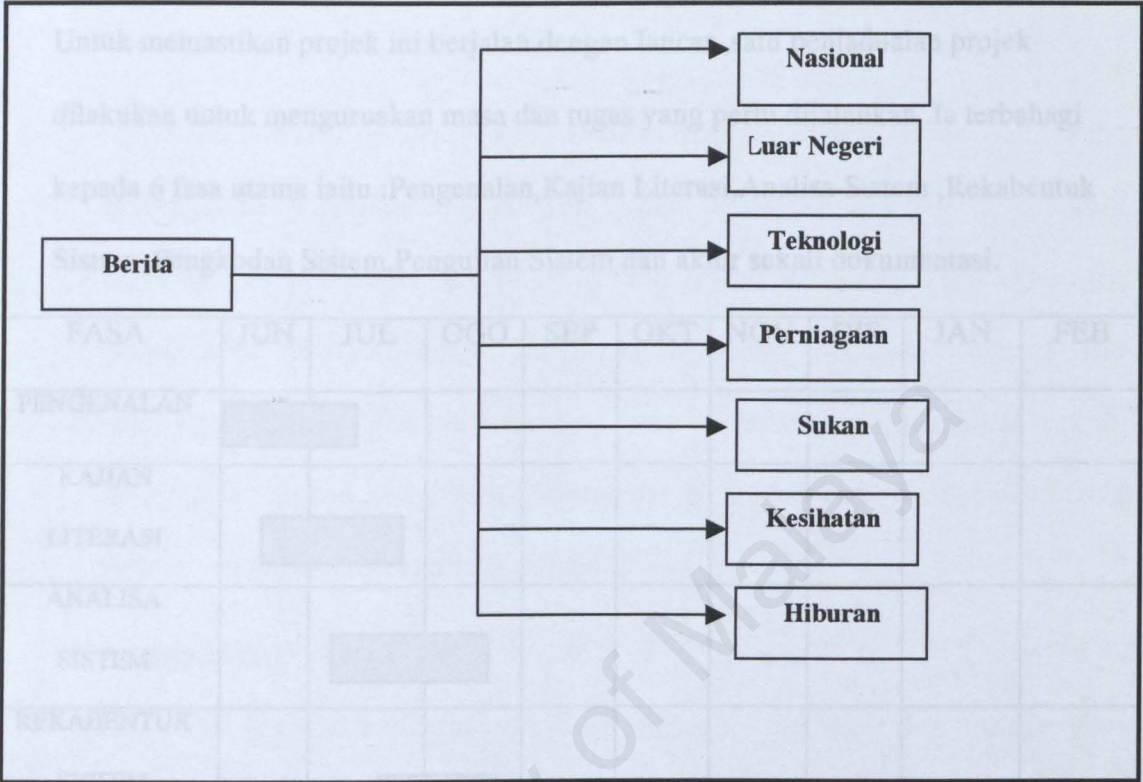
1.3 Objektif Projek

Penetapan objektif adalah penting agar beberapa keperluan sistem dapat diperolehi disamping menyediakan hala tuju dan garis panduan untuk sistem yang lebih baik.

Objektif pembangunan sistem ini meliputi aspek-aspek yang dibincangkan dibawah:

1. Untuk membangunkan satu sistem yang dapat membantu pengguna untuk mengklasifikan rencana atau suatu dokumen mengikut kategori tertentu.
2. Menghasilkan satu sistem yang menjimatkan masa kepada pengguna .
3. Membangunkan satu sistem yang menggunakan teknik-teknik utama di dalam kepintaran buatan iaitu Nearest Neighbour.

1.4 Kategori-kategori Utama



Rajah 1.1 : Kategori-kategori dalam berita

1.5 Kelebihan

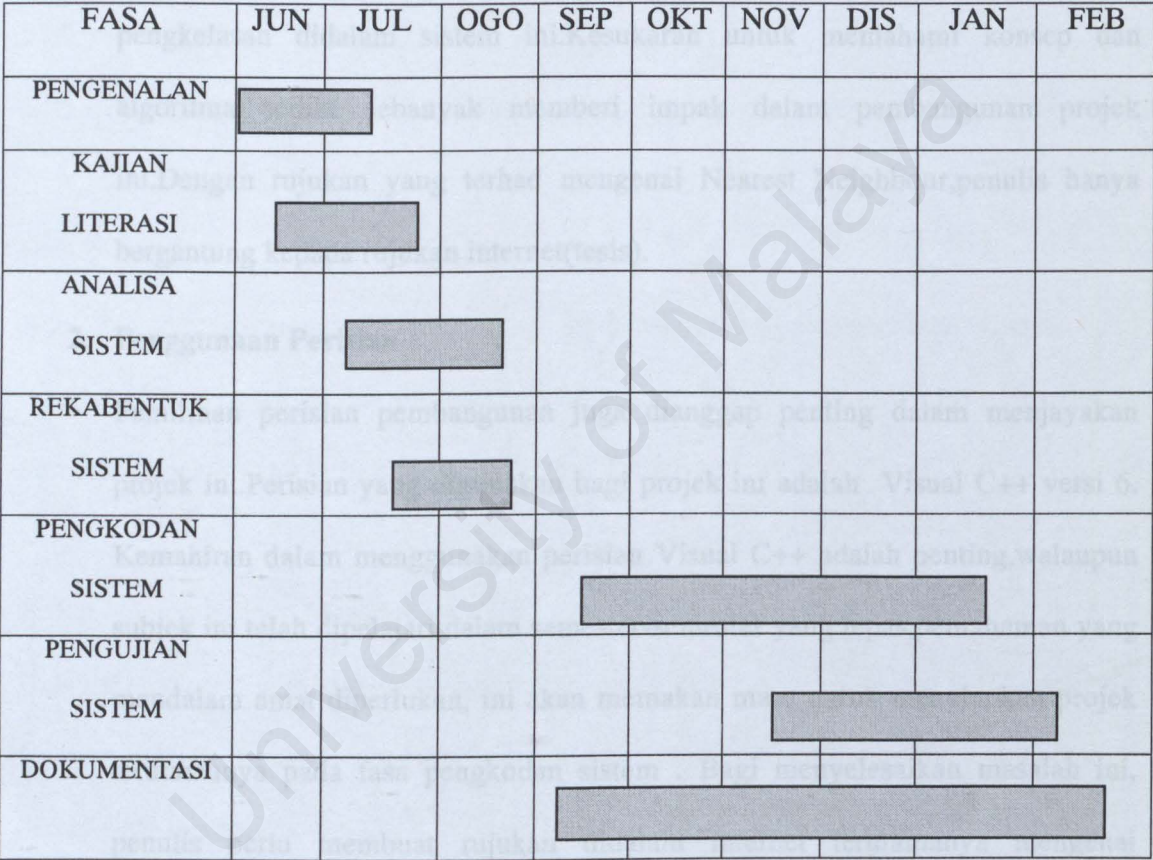
- 1. Dapat menjimatkan masa untuk mengkategorikan artikel-artikel yang dimasukkan

1.6 Sasaran Pengguna

- 1.Pustakawan
- 2.Pelajar

1.7 Jangka Masa Projek

Untuk memastikan projek ini berjalan dengan lancar ,satu penjadualan projek dilakukan untuk menguruskan masa dan tugas yang perlu dijalankan .Ia terbahagi kepada 6 fasa utama iaitu :Pengenalan,Kajian Literasi,Analisa Sistem ,Rekabentuk Sistem ,Pengkodan Sistem,Pengujian Sistem dan akhir sekali dokumentasi.



Jadual 1.1 : Carta Gant Projek

1.8 Mengenalpasti Masalah

1. Pemahaman Teknik Yang Dipilih

Untuk menjayakan projek ini, pemahaman dan pemilihan teknik yang tepat adalah penting. Dalam kajian yang telah dilakukan didapati satu teknik pembelajaran mesin iaitu Nearest Neighbour adalah sesuai untuk dijadikan agen pengelasan didalam sistem ini. Kesukaran untuk memahami konsep dan algoritma sedikit sebanyak memberi impak dalam pembangunan projek ini. Dengan rujukan yang terhad mengenai Nearest Neighbour, penulis hanya bergantung kepada rujukan internet (tesis).

2. Penggunaan Perisian

Pemilihan perisian pembangunan juga dianggap penting dalam menjayakan projek ini. Perisian yang digunakan bagi projek ini adalah Visual C++ versi 6. Kemahiran dalam menggunakan perisian Visual C++ adalah penting, walaupun subjek ini telah dipelajari dalam semester-semester yang lepas, pemahaman yang mendalam amat diperlukan, ini akan memakan masa untuk menyiapkan projek terutamanya pada fasa pengkodan sistem. Bagi menyelesaikan masalah ini, penulis perlu membuat rujukan didalam internet terutamanya mengenai pembangunan antaramuka pengguna.

3. Peruntukan Masa

Peruntuk masa bagi pembangunan projek ini adalah terhad. Jika terdapat fasa-fasa yang melebihi masa yang telah ditetapkan, ini akan mencatatkan pembangunan projek ini. Jadi segala aktiviti-aktiviti kerja mestilah dilakukan mengikut penjadualan yang telah ditetapkan.

1.9 Struktur Laporan

Tujuan laporan ini adalah untuk mendokumentasikan maklumat-maklumat yang terkumpul semasa pembangunan sistem. Laporan ini mengandungi 6 bab (termasuk pengenalan dan kesimpulan)

Bab 1 : Pengenalan

Bab ini adalah pengenalan dan definisi keseluruhan kepada projek dalam bentuk umum.

Bab 2 : Kajian Literasi

Bab ini membincangkan kajian yang telah dilakukan semasa fasa analisis dan rekabentuk projek ini. Kajian ini termasuklah teknik, perisian dan idea yang diperlukan untuk membangunkan sistem Pengkategorian Teks.

Bab 3 : Metodologi

Bab 3 menerangkan kaedah yang digunakan untuk mengumpul data dan maklumat bagi rujukan projek ini. Juga mempertimbangkan model-model yang sedia ada untuk pembangunan projek.

Bab 4 : Rekabentuk Sistem

Beberapa aspek seperti rekabentuk antaramuka, enjin pemprosesan dan spesifikasi keperluan sistem akan dibincangkan didalam bab ini.

Bab 5 : Implementasi

Dalam fasa pembangunan, beberapa aktiviti akan dijalankan seperti pembinaan dan pengujian, pemasangan dan pengujian pakej perisian, penulisan aturcara dan pengujian serta penyediaan dokumentasi. Di dalam aktiviti implementasi pula, aktiviti yang paling utama ialah aktiviti penukaran kepada sistem yang baru .

Bab 7 : Penyelenggaraan

Penyelenggaraan merupakan satu proses di mana aktiviti-aktiviti perubahan yang dilakukan ke atas sistem ditangani dengan memperkenalkan penyelenggaraan yang sesuai. Perubahan-perubahan tersebut perlu dilakukan disebabkan oleh beberapa factor seperti pembetulan, penyesuaian, penyempurnaan dan pencegahan.

Bab 8 : Pembangunan Masa Hadapan

Memberi pendapat untuk membangunkan system yang sedia ada

Bab 9 : Kesimpulan

Kesimpulan keseluruhan projek.

BAB 2 : KAJIAN LITERASI

2.1 Pengenalan Kepada Kajian Literasi

Kajian literasi merupakan kajian untuk mengumpul matlumat yang perlu dibuat dalam usaha untuk membangunkan sistem Pengkategorian Teks(PT).Pengumpulan dibuat dengan menyeluruh untuk membangunkan agar dapat menjadi garis panduan dalam merangka proses pembangunan aplikasi tersebut.

Pengumpulan yang dibuat boleh dibahagikan kepada beberapa kaedah diantaranya ialah

❖ Maklumat daripada sumber internet

Ini merupakan sumber utama dan termudah serta pantas dalam mendapatkan maklumat berkaitan dengan pengkategorian teks.Cara untuk mencari bahan rujukan ini adalah dengan menggunakan enjin pencarian seperti Yahoo,Google dan Altavista.

❖ Perpustakaan

Merujuk daripada buku-buku teks,jurnul-jurnul,surat khabar,majalah dan lain-lain yang berkaitan dengan matlumat untuk membangunkan aplikasi ini.

2.2 Pengenalan Kepada Pengkategorian Teks

Definasi :-Pengkategorian teks – mengkategorikan teks bahasa semulajadi secara automatik ke kategori-kategori tertentu berdasarkan kepada kandungan dokumen. Ia adalah aplikasi termasuk mengindekkan teks yang menyokong mengeluarkan dokumen, mengeluarkan data dari teks.Satu aplikasi primari sistem pengkategorian teks yang menentukan kategori subjek kepada dokumen untuk menyokong pengambilan matlumat (information retrieval), atau membantu seseorang pengindeks menentukan kategori tertentu. Komponen pengkategorian teks dilihat meningkatkan lagi penggunaan sistem pemprosesan bahasa semulajadi untuk mengeluarkan data . Pengkategorian teks ini juga dilihat sebagai satu komponen penapis untuk sesetengah bidang.

Sistem pengkategorian teks cuba untuk menghasilkan semula penilaian pengkategorian manusia.Satu pendekatan untuk membina sistem pengkategorian teks ini ialah secara manual menempatkan sesetengah set dokumen ke kategori,dan kemudian menggunakan pembelajaran induktif untuk menentukan secara automatik kategori berdasarkan kandungan perkataan .

Terdapat beberapa aplikasi Pengkategorian Teks iaitu :

Halaman Web

- Mencadangkan (Recommending)
- Pengkelasan Yahoo

2.3 Contoh Berita Yang Belum Diproses

Carrier confident of improving market position

Newsgroup Messages

DESPITE aggressive inroad by Chinese and South Korean air conditioning brands, Carrier (Malaysia) Sdn Bhd is confident of improving its market position with the advent of Afta next year, through high-quality product offerings and aggressive sales and marketing activities within the Asian region.

- Mencadangkan (Recommending)
- Penapisan spam

Carrier is among the top 5 air conditioning company in Malaysia and its manufacturing division, Carrier International Sdn Bhd (CISB) which is located in Hong Kong, currently export about 70 per cent of its products to West Asia, Thailand, Vietnam, Singapore, Philippines, Australia, Hongkong and Indonesia.

Artikel Akhbar

Rajah 2.1 : Contoh berita

- Mengkelaskan mengikut kategori-kategori tertentu

Soccer: Indra's switch to Syed Ahmad in the making

Masej Email

Indra, 21, voted the most promising footballer in the Perak FA, plays in a similar position as Syed Ahmad did in the 70s. The lucky player, given an attacking midfielder's role by former coach Karl Weigand, has been transformed into a top striker by present coach Tony Brown. He will fill in for Khalid Jamius, who is in Germany with a Division Two club.

- Routing
- Prioritizing
- Folderizing
- Penapisan spam

The switch has been a success for Indra who is on a hot streak and relishing his new role. He has developed a strong appetite for goals.

Syed was noted for his speed and skill and Indra, who is having a fine run this season, displays uncanny similarities with the former striker. What he lacks in speed, he more than makes up for with his shooting skills.

Indra has been a constant goal-getter for Perak since filling in for Khalid. He scored a goal each in all the FA Cup matches this season as well as in the three Malaysia Cup matches. It was his goals that gave Perak full points against NS Champake, Johor FA and Kedah.

Rajah 2.2 : Contoh berita

2.3 Contoh Berita Yang Belum Diproses

Carrier confident of improving mart position

DESPITE aggressive inroad by Chinese and South Korean air conditioning brands, Carrier (Malaysia) Sdn Bhd is confident of improving its market position with the advent of Afta next year, through high-quality product offering, and competitive product sourcing from its factories within the Asean region.

Carrier is among the top three air conditioning company in Malaysia and its manufacturing division, Carrier International Sdn Bhd (CISB) which is located in Bangi, currently export about 70 per cent of its products to West Asia, Thailand, Vietnam, Singapore, Philippines, Australia, Hongkong and Indonesia.

Rajah 2.1 : Contoh berita

Soccer: Indra another Syed Ahmad in the making

Indra, 21, voted the most promising footballer by the Perak FA, plays in a similar position as Syed Ahmad did in the 70s. The lanky player, given an attacking midfielder's role by former coach Karl Weigang, has been transformed into a top striker by present coach Tony Netto and will fill in for Khalid Jamlus, who is in Germany with a Division Two club.

The switch has been a blessing in disguise for Indra who is on a hot streak and relishing his new role. He has developed a healthy appetite for goals.

Syed was noted for his speed and skill and Indra, who is having a fine run this season, displays uncanny similarities with the former striker. What he lacks in speed, he more than makes up for with his shooting skills.

Indra has been a constant goal-getter for Perak since filling in for Khalid. He scored a goal each in all the FA Cup matches this season as well as in the three Malaysia Cup matches. It was his goals that gave Perak full points against NS Chempaka, Johor FA and Kedah.

Rajah 2.2 : Contoh berita

2.4 Pengenalan Kepada Pemprosesan Bahasa Tabii

Three-A makes Mesdaq mart debut with 2.5-sen premium

FOOD and beverage ingredients manufacturer Three-A Resources Bhd recorded a closing premium of 10 per cent or 2.5 sen to 27.5 sen on its maiden day on the Mesdaq Market yesterday.

A total of 6.3 million shares were traded. The counter opened at 29 sen on 1.485 million shares and hit its high of 30 sen and low of 27.5 sen.

Commenting on the opening premium, executive chairman Datuk Mohd Nor Abdul Wahid said: "The premium is very good."

Three-A manufactures over 17 different varieties of caramel colour, HVP, natural fermented vinegar (NFV) and glucose/maltose syrup and exports about 30 per cent of these products to 13 countries which include Hong Kong, Taiwan, China, Indonesia, Myanmar, the Philippines, Singapore, the US, Australia, Poland, Vietnam, Cambodia and Thailand.

Rajah 2.3 : Contoh berita

- Mencari dokumen yang mengandungi topik tertentu daripada pengkalan data bagi contoh mencari buku berkaitan di perpustakaan.
- Mengeluarkan maklumat daripada artikel bagi topik tertentu. Contoh membina pengkalan data bagi urusan stok.
- Menterjemah dokumen daripada satu bahasa ke bahasa yang lain.
- Membuat kesimpulan bagi sesuatu artikel.

2.4 Pengenalan Kepada Pemprosesan Bahasa Tabii

Didalam membangunkan Pengkategorian Teks, Pemprosesan Bahasa Tabii adalah merupakan salah satu sumber rujukan utama. Ia adalah sebahagian daripada aktiviti yang diperlukan untuk membina atau membentuk sesuatu sistem yang bergantung kepada penggunaan bahasa tabii.

Objektif utama Pemprosesan Bahasa Tabii ialah bagi merekabentuk dan membina sistem komputer yang boleh menganalisa, memahami dan menjanakan bahasa tabii manusia. Aplikasi bagi Pemprosesan Bahasa Tabii boleh dibahagikan kepada 2 kelas utama iaitu :

- ❖ Text-Based Applications
- ❖ Dialogue-based applications

Text-Based Applications

Melibatkan pemprosesan teks bertulis seperti buku, suratkhbar, manual, email dan sebagainya. Contoh aplikasi ialah seperti :

- Mencari dokumen yang mengandungi topik tertentu daripada pengkalan data bagi teks. Contoh mencari buku berkaitan di perpustakaan.
- Mengeluarkan matlumat daripada artikel bagi topik tertentu. Contoh membina pengkalan data bagi urusan stok
- Menterjemah dokumen daripada satu bahasa ke bahasa yang lain
- Membuat kesimpulan bagi sesuatu artikel.

Dialogue-based applications

Melibatkan komunikasi manusia-mesin seperti bahasa pertuturan dan juga interaksi melalui papan kekunci.

Contoh-contoh aplikasi ialah :

- Question-answering system dimana bahasa tabii digunakan sebagai query. Contoh query system bagi pengkalan data personal.
- Automated customer service melalui telefon.
- Intelligent Tutoring System dimana mesin berinteraksi dengan pelajar.

Terdapat beberapa pengetahuan yang berkaitan dengan pemahaman bahasa tabii adalah seperti berikut:-

1. Pengetahuan phonetik dan phonologikal

Pengetahuan yang menunjukkan bagaimana perkataan boleh dikaitkan dengan bunyi atau lebih jelas berkenaan dengan ejaan yang mengikut sebutan.

2. Pengetahuan morfologikal

Pengetahuan yang menunjukkan bagaimana beberapa perkataan dapat dihasilkan daripada suatu perkataan dasar. Contohnya perkataan akar ditambah dengan imbuhan 'ber-' lalu menjadi 'bermain'.

3. Pengetahuan sintaktiks

Pengetahuan yang menunjukkan bagaimana beberapa perkataan digabungkan untuk membentuk suatu ayat atau frasa yang betul.

4. Pengetahuan semantiks

Pengetahuan yang menunjukkan apakah makna suatu perkataan dan bagaimana makna suatu perkataan itu boleh membentuk suatu ayat yang bermakna.

5. Pengetahuan pragmatik

Pengetahuan yang menunjukkan bagaimana suatu ayat digunakan pada situasi yang berbeza dan bagaimana penggunaannya dapat mempengaruhi pentafsiran suatu ayat.

6. Pengetahuan Leksikel

Pengetahuan yang menunjukkan bagaimana ayat awalan mempengaruhi pentafsiran pada ayat seterusnya.

7. Pengetahuan World

Pengetahuan yang melibatkan dunia fizikal contohnya interaksi sosial antara manusia. Pengetahuan ini penting untuk memahami makna sebenar pada suatu teks atau pertuturan.

2.5 Algoritma Nearest Neighbour

Terdapat beberapa pengkelasan utama di dalam pembelajaran mesin iaitu Naive Bayes Classifier , Decision Tree, Neural Network, Support Vector Machine, Least Linear Square Fit, Boosting and Nearest Neighbor Classifier. Naïve Bayes, Rocchio, Probabilistic Indexing

Di dalam projek ini algoritma Nearest Neighbor telah dipilih untuk digunakan didalam menyelesaikan masalah dalam membangunkan aplikasi ini . Algoritma ini telah dibuktikan efektif dan secara meluas digunakan dalam pelbagai penyelesaian masalah dan juga ia merupakan algoritma yang mudah difahami . Nearest neighbor menganggar nilai dengan setkan nilai titik data terdekat (dengan menggunakan matrik jarak(distance metric)).

Perkara asas di dalam k-nearest neighbor adalah seperti berikut :

- Menyimpan semua instance, termasuk klasifikasinya
- Untuk instance yang baru , kira jarak di antara instance yang baru dengan setiap instance yang tersimpan .

2.5.1 Algoritma K-Nearest Neighbor

K-nearest neighbor dikenali sebagai pendekatan statistik dimana telah dikaji dalam pencaman corak dan digunakan untuk pengkategorian teks . Secara umum , algoritma K-nearest neighbor akan mencari instance k-terdekat ke query instance dan memastikan kelasnya dengan mengenalpasti salah satu kelas yang berhubung-kait dengan jiran terdekat. Nilai bagi k sebenarnya akan memberi kesan kepada prestasi algoritma.

Terdapat 2 kaedah utama untuk membuat ramalan dokumen latihan: undian majoriti dan kesamaan penjumlahan markah (Similarity Score Summing). Didalam latihan undian, sesuatu kategori hanya mendapat satu undian untuk setiap instance kategori itu dalam set k susunan teratas nearest neighbors. Bagaimanapun kategori persamaan adalah yang mendapatkan markah tertinggi pengundian. Setiap kategori mendapat markah yang sama kepada jumlah markah kesamaan instance kategori itu dalam k susunan-teratas nearest neighbors. Kategori yang hampir sama adalah yang tertinggi jumlah persamaan markah. Dalam kata lain, jika jarak adalah kurang di antara 2 instance didalam ruang, maka lebih persamaan diantara mereka. Persamaan markah setiap dokumen jiran kepada dokumen diklasifikasi menggunakan pemberat untuk setiap kategori, dan jumlah kategori pemberat melebihi k -nearest neighbor digunakan untuk menyusun kategori. Persamaan nilai di antara 2 instance adalah jarak di antara ia berdasarkan kepada jarak matrik. Secara umum, Jarak Matrk Eucladian yang biasa digunakan.

Pengkelasan k -NN adalah asas kepada kebanyakan algoritma pembelajaran dan ia adalah pasti yang k -NN adalah purely lazy. Algoritma pembelajaran purely lazy secara umum dicirikan dengan 3 tingkahlaku dibawah:

1. Perbezaan (*Defer*). Ia menyimpan semua data yang dilatih dan membezakan pemprosesan sehingga queri diberikan perlu memberi tindakbalas (*reply*).

2. Memberi Tindakbalas (*Reply*). Quer dijawab dengan menggabungkan data yang dilatih, biasanya menggunakan pendekatan pembelajaran tempatan (*local*)

learning approach) dimana (i)instance definisikan sebagai titik didalam ruang,(ii)fungsi persamaan definisikan ke atas semua pasangan instance,(iii) fungsi ramalan definisikan jawapan menjadi fungsi monotik persamaan queri.

3.Pembuangan (*flush*). Selepas mengenakan kepada queri,jawapan dan keputusan pertengahan akan dibuang.

Kesimpulannya , boleh dikatakan k -NN menyimpan keseluruhan set latihan dan tangguhkan semua usaha ke arah generelasi induktif sehingga masa pengelasan.

Penguasaan proses datang dari proses pengambilan .Diberikan instance ujian baru,cari kes latihan yang disimpan dimana ia yang terdekat berdasarkan pada pengukuran jarak, ambil kelas kes pengambilan, dan ramalkan instance baru yang akan berada didalam kelas yang sama.

Training:

- [1] $\forall x_t \in \text{Training Set}$
- [2] Store x_t in memory

Querying:

- [1] $\forall x_q \in \text{Query Set}$
- [2] $\forall x_t \{x_t \neq x_q\}$: Calculate $\text{Similarity}(x_q, x_t)$
- [3] Let *Similar* be set of k most similar instances to x_q in Training Set
- [4] Let $\text{Sum} = \sum_{x_t \in \text{Similar}} \text{Similarity}(x_q, x_t)$
- [5] Then return the categories of instances in Similar, in decreasing order by the number of times the category is seen in Similar.

Rajah 2.4 : k -NN Regression

2.5.2 Matrik Jarak

Algoritma nearest neighbour menggunakan matrik untuk mengukur jarak di antara instance baru dengan set instance yang ada didalam ingatan. Fungsi jarak digunakan untuk memastikan persamaan atau perhubungan jarak di antara dua instance. Instance yang baru akan diklasifikasikan berdasarkan nearest neighbour. Matrik jarak perlu memenuhi kreteria dibawah:

- $D(i, j) > 0$ if $i \neq j$
- $D(i, j) = D(j, i)$
- $D(i, j) = 0$ jika hanya $i = j$
- $D(i, j) + D(j, k) \geq D(i, k)$

Dimana $D(i, j)$ adalah jarak diantara dua instance iaitu i dan j .

Pemilihan matrik bergantung kepada nilai atribut dimana ia boleh diwakili samada dengan nilai atau dengan simbol.

2.6 Pemprosesan Indeks

Untuk menghasilkan pengkategorian yang efektif dan berkesan , ia mestilah mencari matlumat daripada teks dalam dokumen tersebut . Beberapa proses telah dikenalpasti untuk mengekstrakkan teks dalam dokumen tersebut . Proses-proses digabungkan menjadi satu proses utama dipanggil pengindeksan dokumen .

Di dalam pengkategorian Teks proses indek adalah proses yang paling penting sekali dimana ia mengatur dokumen teks berdasarkan kepada kandungannya.Ia dijalankan dengan meletakkan setiap dokumen dengan kata kunci mewakili dokumen Asas automatik pengindeksian ini dilakukan dengan sistem itu sendiri memastikan teks di dalam dokumen .Langkah-langkah adalah seperti berikut :

1. Hapuskan Perenggan
2. Hapuskan koma,titik,hypen
3. Saperator Indicator
4. Hapuskan Nombor,Tanda Mata Wang
5. Menukarkan huruf besar ke huruf kecil
6. Hapus Stopword
7. Word Stemming
8. Pengekstrakan Maklumat

Rajah 2.5 : Hapuskan Perenggan

2.6.1 Hapuskan Perenggan

Sistem akan menghapuskan perenggan yang ada di dalam setiap rencana. Langkah

Seperti berikut :

Indra 21 voted the most promising footballer by the Perak FA, plays in a similar position as Syed Ahmad did in the 70s. The lanky player, given an attacking midfielder's role by former coach Karl Weigang, has been transformed into a top striker by present coach Tony Netto and will fill in for Khalid Jamlus, who is in Germany with a Division Two club.

The switch has been a blessing in disguise for Indra who is on a hot streak and relishing his new role. He has developed a healthy appetite for goals.

Syed was noted for his speed and skill and Indra, who is having a fine run this season, displays uncanny similarities with the former striker. What he lacks in speed, he more than makes up for with his shooting skills.



Indra, 21, voted the most promising footballer by the Perak FA, plays in a similar position as Syed Ahmad did in the 70s. The lanky player, given an attacking midfielder's role by former coach Karl Weigang, has been transformed into a top striker by present coach Tony Netto and will fill in for Khalid Jamlus, who is in Germany with a Division Two club. The switch has been a blessing in disguise for Indra who is on a hot streak and relishing his new role. He has developed a healthy appetite for goals. Syed was noted for his speed and skill and Indra, who is having a fine run this season, displays uncanny similarities with the former striker. What he lacks in speed, he more than makes up for with his shooting skills.

Rajah 2.5 : Hapuskan Perenggan

2.6.2 Hapuskan titik,koma,hypen

Pada umumnya titik dan koma akan dibuang .Tanda hypen '-' juga dibuang semasa proses pengindeksan.

Indra 21 voted the most promising footballer by the Perak FA plays in a similar position as Syed Ahmad did in the 70s The lanky player given an attacking midfielder's role by former coach Karl Weigang has been transformed into a top striker by present coach Tony Netto and will fill in for Khalid Jamlus who is in Germany with a Division Two club The switch has been a blessing in disguise for Indra who is on a hot streak and relishing his new role He has developed a healthy appetite for goals Syed was noted for his speed and skill and Indra who is having a fine run this season displays uncanny similarities with the former striker What he lacks in speed he more than makes up for with his shooting skills

Rajah 2.6 : Hapuskan titik,koma,hypen

2.6.3 Saperator Indicator

Selepas titik dijumpai di dalam sesuatu ayat, pembahagi ' / ' akan diletakkan di antara setiap ayat.Contoh seperti dibawah :

Indra voted the most promising footballer by the Perak FA plays in a similar position as Syed Ahmad did in the / The lanky player given an attacking midfielder's role by former coach Karl Weigang has been transformed into a top striker by present coach Tony Netto and will fill in for Khalid Jamlus who is in Germany with a Division Two club/The switch has been a blessing in disguise for Indra who is on a hot streak and relishing his new role/ He has developed a healthy appetite for goals/ Syed was noted for his speed and skill and Indra who is having a fine run this season displays uncanny similarities with the former striker/ What he lacks in speed he more than makes up for with his shooting skills/

Rajah 2.7 : Saperator Indicator

2.6.4 Hapuskan Nombor Dan Simbol Istimewa

Nombor dan symbol istimewa seperti simbol mata wang akan dibuang.

Stopword adalah perkataan yang selalu digunakan dan mempunyai nilai maklumat yang

Indra voted the most promising footballer by the Perak FA plays in a similar position as Syed Ahmad did in the The lanky player given an attacking midfielders role by former coach Karl Weigang has been transformed into a top striker by present coach Tony Netto and will fill in for Khalid Jamlus who is in Germany with a Division Two clubThe switch has been a blessing in disguise for Indra who is on a hot streak and relishing his new role He has developed a healthy appetite for goals Syed was noted for his speed and skill and Indra who is having a fine run this season displays uncanny similarities with the former striker What he lacks in speed he more than makes up for with his shooting skills

Rajah 2.8 : Hapuskan Nombor Dan Simbol Istimewa

2.6.5 Menukar Huruf Besar Ke Huruf Kecil

Kesemua huruf besar akan ditukarkan kepada huruf kecil

indra voted the most promising footballer by the perak fa plays in a similar position as syed ahmad did in the / the lanky player given an attacking midfielders role by former coach karl weigang has been transformed into a top striker by present coach tony netto and will fill in for khalid jamlus who is in germany with a division two club/the switch has been a blessing in disguise for indra who is on a hot streak and relishing his new role/ he has developed a healthy appetite for goals/ syed was noted for his speed and skill and indra who is having a fine run this season displays uncanny similarities with the former striker/ what he lacks in speed he more than makes up for with his shooting skills/

Rajah 2.9 : Menukar Huruf Besar Ke Huruf Kecil

Stopword adalah perkataan yang selalu digunakan dan mempunyai nilai matlumat yang sedikit .Perkataan ini tidak akan dicari di dalam dokumen.Perkataan ini penting kerana jika selalu digunakan di dalam sesuatu ayat , ia akan semakin tidak berguna sebagai satu kunci kata yang unik.Contoh for, in , a dan lain-lain.

2.6.6 Hapus Stopword

Stopword adalah perkataan yang selalu digunakan dan mempunyai nilai matlumat yang sedikit. Perkataan ini tidak akan dicari di dalam dokumen. Perkataan ini penting kerana jika selalu digunakan di dalam sesuatu ayat, ia akan semakin tidak berguna sebagai satu kunci kata yang unik. Contoh for, in, a dan lain-lain.

a	can	hereafter	most	quickly	then	which
about	cannot	hereby	mostly	quite	thence	while
above	cc	herein	mr	rather	there	whither
abs	cm	hereupon	much	readily	thereafter	who
accordingly	come	hers	mug	really	thereby	whoever
across	compare	herself	must	recently	therefore	whom
after	could	him	my	refs	therein	whose
afterwards	de	himself	myself	regarding	thereupon	why
again	dealing	his	namely	relate	these	will
against	department	how	nearly	said	they	with
all	depend	however	necessarily	same	this	within
almost	did	hr	neither	seem	thorough	without
alone	discover	i	never	seemed	those	wk
along	dl	ie	nevertheless	seeming	though	would
already	do	if	next	seems	through	wt
also	does	ii	no	seen	throughout	yet
although	done	iii	nobody	seriously	thru	you
always	due	immediately	noone	several	thus	your
am	during	importance	nor	shall	to	yours
among	each	important	normally	she	together	yourself
amongst	ec	in	nos	should	too	yourselves
an	ed	inc	noted	show	toward	yr
analyze	effected	incl	nothing	showed	towards	
and	eg	indeed	now	shown	try	
another	either	into	nowhere	shown	type	
any	else	investigate	obtained	shows	ug	
anyhow	elsewhere	is	of	significantly	under	
anyone	enough	it	off	since	unless	
anything	especially	its	often	slightly	until	
anywhere	et	itself	on	so	up	
applicable	etc	just	only	some	upon	
apply	ever	keep	onto	somehow	us	
are	every	kept	other	someone	use	
arise	everyone	kg	others	something	used	
around	everything	km	otherwise	sometime	usefully	
as	everywhere	last	ought	sometimes	usefulness	
assume	except	latter	our	somewhat	using	
at	find	latterly	ours	somewhere	usually	
be	for	lb	ourselves	soon	various	
became	found	ld	out	specifically	very	
because	from	letter	over	still	via	
become	further	like	overall	strongly	was	
becomes	gave	ltd	owing	studied	we	
becoming	get	made	own	sub	were	
been	give	mainly	oz	substantially	what	
before	go	make	particularly	such	whatever	
beforehand	gone	many	per	sufficiently	when	
being	got	may	perhaps	take	whence	
below	gov	me	pm	tell	whenever	
beside	had	meanwhile	precede	th	where	
besides	has	mg	predominantly	than	whereafter	
between	have	might	present	that	whereas	
beyond	having	ml	presently	the	whereby	
both	he	mm	previously	their	wherein	
but	hence	mo	primarily	theirs	whereupon	
by	her	more	promptly	them	wherever	
came	here	moreover	pt	themselves	whether	

Jadual 2.1 : Senarai stopword

2.6.7 Word Stemming

Dalam sesetengah kes, morfologikal varian perkataan mempunyai interpretasi semantic yang sama dan boleh dianggap sama bagi tujuan aplikasi pencapaian matlumat (Information Retrieval). Bagi tujuan ini, satu algoritma *stemming*, atau *stemmers*, telah dibangunkan, dimana ia akan cuba untuk mengurangkan atau membuang kata imbuhan kepada perkataan dasar. Perkataan dasar yang dihasilkan didapati dengan membuang affix daripada perkataan. Affix adalah elemen verbal yang diletakkan di permulaan perkataan (prefix), di tengah-tengah perkataan (infix) dan diakhir perkataan (suffix).

2.6.8 Pengekstrakan Maklumat

Terdapat beberapa perkataan boleh dipilih sebagai kata kunci yang merujuk kepada kandungan dokumen. Pengecaman ini adalah untuk beberapa konsep spesifik contohnya seperti orang, organisasi, tempat, produk, peralatan, kenderaan, peristiwa dan lain-lain. "Meta" seperti #COMPANY, #PERSON ditambah untuk pengindeksan. Pencaman boleh ditambah bagi memudahkan proses pengindeksan.

Terdapat beberapa masalah utama dalam proses ini, salah satu ialah

- sukar untuk dibina
- masalah ketepatan
- mendapatkan konsep yang betul

Beberapa pengecaman dikenalpasti untuk proses pengindeksan :

Pengecaman negara : Untuk setiap negara yang dinyatakan didalam teks ,ia akan menghasilkan pengecaman untuk #COUNTRY.Contoh america-> #COUNTRY america .

Pengecaman negari : Untuk setiap negari yang dinyatakan didalam teks ,ia akan menghasilkan pengecaman untuk #STATE.Contoh kedah-> #STATE kedah .

2.7 Analisa Sistem Yang Sedia Ada

Pengecaman tempat : Untuk setiap tempat yang dinyatakan didalam teks ,ia akan menghasilkan pengecaman untuk #PLACE.Contoh field-> # PLACE field .

Pengecaman tindakan : Untuk setiap tindakan yang dinyatakan didalam teks ,ia akan menghasilkan pengecaman untuk #ACTION.Contoh invest-> #ACTION invest.

Pengecaman organisasi: Untuk setiap organisasi yang dinyatakan didalam teks ,ia akan menghasilkan pengecaman untuk #ORGANIZATION . Contoh Red Cross -> #ORGANIZATION Red Cross .

Pengecaman peralatan: Untuk setiap peralatan yang dinyatakan didalam teks ,ia akan menghasilkan pengecaman untuk #EQUIPMENT . Contoh computer-> #EQUIPMENT computer.

Pengecaman kenderaan: Untuk setiap produk yang dinyatakan didalam teks ,ia akan menghasilkan pengecaman untuk #VEHICLE.Contoh motorcycle-> #PRODUCT motorcycle.

Pengecaman peristiwa: Untuk setiap negara yang dinyatakan didalam teks ,ia akan menghasilkan pengecaman untuk #EVENT.Contoh charity-> #EVENT charity.

2.7 Analisa Sistem Yang Sedia Ada

Analisa sistem sedia ada dijalankan untuk mengkaji sejauh mana keberkesanan sistem tersebut dalam mencapai objektif yang telah digariskan.Dalam konteks ini,kajian dijalankan terhadap beberapa jenis pengkategorian teks yang mempunyai objektif yang sama tetapi dibagunkan dengan kaedah-kaedah yang berlainan den mempunyai kelebihan serta kekurangan masing-masing.

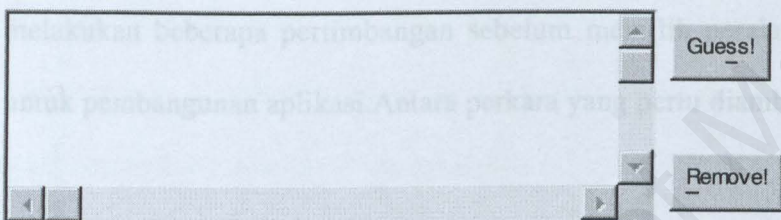
2.7.1 Pengenalan Kepada Sistem Pengkategorian Teks Yang Ada

2.7.2 TEXTCAT

TextCat adalah implementasi algoritma pengkategorian teks yang dipersembahkan oleh Cavnar, W. B. and J. M. Trenkle, dengan menggunakan N-Gram-Based algoritma. Algoritma ini diimplementasikan untuk membangunkan aplikasi yang dapat mengenalpastikan bahasa yang digunakan.. Aplikasi yang dibangunkan dapat mengenalpastikan sebanyak 69 bahasa babii.

Dibawah ini adalah demonstrasi Peneka Bahasa yang dibangunkan berdasarkan , **N-Gram-Based Text Categorization**. Dibangunkan menggunakan perisian pembangunan Perl.

Taipkan beberapa baris perkataan. Lebih banyak perkataan yang dapat dimasukkan lebih baik peneka bahasa berfungsi.



RESULT

english

Rajah 2.10 : Rajah Antaramuka TextCat

2.8 Perisian Pembangunan Aplikasi

Pemilihan perisian yang sesuai adalah penting dalam melancarkan pelaksanaan projek ini. Penggunaan perisian membantu pengaturcara dan pembangun sistem untuk membangunkan sebuah sistem dengan lebih baik, cekap, tepat, berkualiti dan dalam kos yang lebih murah dan munasabah.

Dengan itu adalah penting untuk seseorang pembangun aplikasi untuk melakukan beberapa pertimbangan sebelum memilih peralatan yang ingin digunakan untuk pembangunan aplikasi. Antara perkara yang perlu diambil kira termasuklah:

1. Ianya perlulah mudah untuk dipelajari dan digunakan. Ini adalah penting untuk memastikan masa untuk memahami dan mempelajari perisian tersebut adalah minimum dan aplikasi dapat disiapkan dan dihantar pada masa yang ditetapkan.
2. Perisian perlulah mempunyai ciri-ciri yang diinginkan bagi sesebuah aplikasi yang ingin dibangunkan. Sebagai contoh jika sekiranya aplikasi itu memerlukan capaian ke atas pangkalan data, maka perisian yang dipilih perlulah mempunyai keupayaan untuk membuat penyambungan ke pangkalan data.
3. Perisian juga perlulah membenarkan integrasi dengan perisian yang lain. Pengguna masa kini tidak mahu perisian yang terbatas penggunaannya dan tidak mampu berkomunikasi dengan perisian yang lain.

4. Perisian yang dipilih perlulah berupaya untuk melakukan pelbagai jenis kerja pembangunan aplikasi termasuklah menjana kod-kod aturcara dan sesuai untuk pelbagai bahasa pengaturcaraan.

5. Dari sudut pemprototaipan kriteria untuk pemilihan perisian perlulah menjurus kepada kecekapan, kepantasan dan kemudahan untuk penggunaannya.

Perisian perlulah interaktif untuk membimbing pembangun sistem dalam proses pembangunan sistem. Dengan ini perisian akan lebih mudah dijelajahi dan seterusnya mengurangkan masa untuk mempelajarinya

2.8.1 Visual C++

Bahasa pengaturcaraan Visual C++ adalah bahasa yang paling penting dan terkenal diantara bahasa pengaturcaraan hari ini. Dibangunkan oleh Microsoft, Visual C++ adalah versi C++, dan juga bahasa pengaturcaraan C. Asalnya dibangunkan oleh Kernigham dan Ritchie di Bell Telephone Laboratories pada tahun 1970, C telah terbukti popular untuk membangunkan aplikasi sistem seperti sistem operasi, pengkompil dan komunikasi.

Ciri-ciri utama Visual C++ adalah pengaturcaraan objek-oriented (OOP). OOP telah mempertingkatkan kualiti perisian dengan menyediakan ciri-ciri baru iaitu penyembunyian data dan enkapsulasi, perwarisan, polimorfisma, fungsi dan operator muatan lebih dan penerimaan.

BAB 3 : METADOLOGI

3.1 Pengenalan

Proses Kejuruteraan Perisian meliputi set langkah yang merangkumi kaedah , peralatan dan prosedur . Set langkah ini biasanya dirujuk sebagai Paradigma Kejuruteraan Perisian atau model kitar hayat Pembangunan Perisian (SDLC) .Paradigma ini biasanya dipilih berdasarkan sifat projek yang dijalankan dan aplikasi serta peralatan dan prosedur yang akan digunakan.

Paradigma atau model untuk pembangunan sistem yang juga dikenali sebagai metadologi terdiri daripada beberapa bentuk yang mempunyai kekuatan dan kelemahannya yang tersendiri . Pun begitu , setiap model ini mempunyai fasa generic yang sama iaitu fasa definisi , pembangunan dan penyelenggaraan.

Secara amnya , metadologi atau cara untuk melakukan sesuatu mempunyai beberapa objektif yang tertentu dalam pembangunan sistem iaitu:

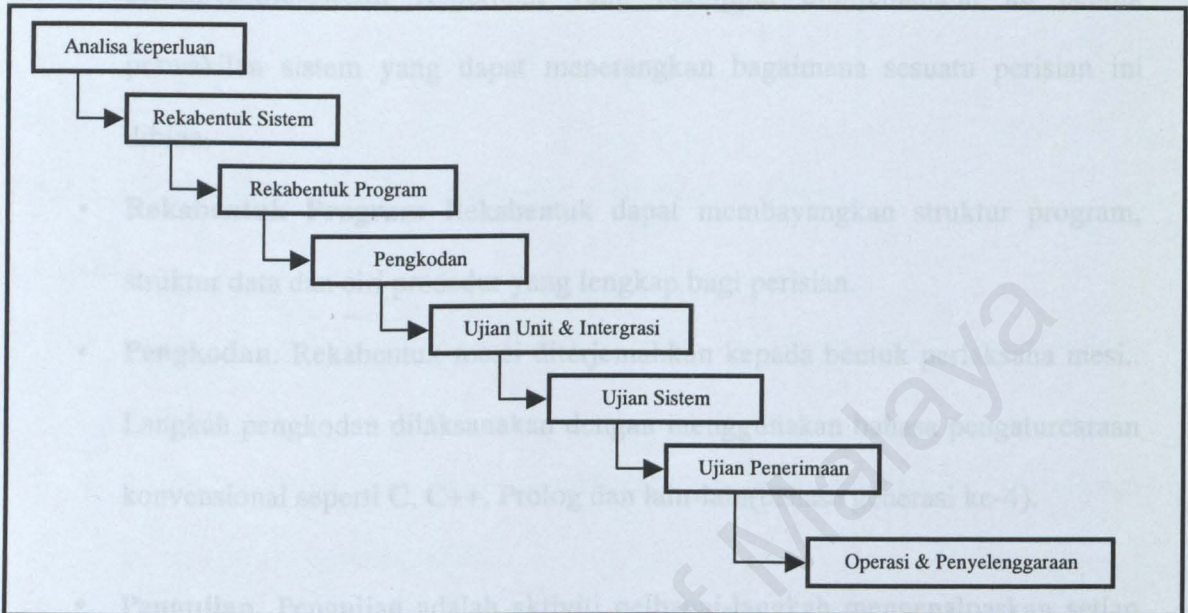
1. Mendapatkan keperluan sistem yang tepat
2. Menyediakan cara yang sistematik untuk membangunkan sistem dengan itu
perkembangan sistem akan dapat diketahui disepanjang proses pembangunannya.

3. Menghasilkan sistem yang boleh didokumentasikan dengan baik dan mudah diselenggarakan.
4. Mengenalpasti sebarang perubahan yang perlu seawal mungkin dalam kitar hayat sistem
5. Menghasilkan sistem yang mempunyai kebolegunaan yang tinggi.
6. Mewujudkan pemahaman yang sama terhadap aktiviti, sumber dan kekangan.
7. Membantu mencari ketidakonsistenan,lewahan di dalam proses.

Model -model yang akan dipertimbangkan di bawah tajuk ini adalah perwakilan terhadap langkah-langkah yang meliputi fasa-fasa dalam pembangunan sistem. Pertimbangan ini adalah penting untuk mengenalpasti kekuatan dan kelemahan yang wujud disepanjang fasa tersebut. Pemilihan metodologi yang bersesuaian dan tepat dapat membantu dalam membangunkan sistem bagi projek ini.

3.2 Pertimbangan Dan Analisis Metodologi

3.2.1 Model Air Terjun



Rajah 3.1 : Model Air Terjun

Pembangunan sistem mengikut model ini adalah cara yang paling mudah dan ringkas. Seperti dalam gambarajah diatas, proses pembangunan mesti disiapkan sebelum ke fasa yang seterusnya. Oleh itu semua keperluan yang diperlukan oleh pengguna, analisa untuk konsistensi, dan mendokumen dokumen yang diperlukan kemudian pembangun akan melakukan aktiviti rekabentuk sistem. Model ini memudahkan untuk melihat apa yang berlaku semasa proses pembangunan sistem.

Sistem dibangunkan mengikut versi. Versi yang siap akan diserahkan kepada pengguna untuk dibuat penilaian. Kemudian, sistem akan menjalani penyelenggaraan sehinggalah ianya bersedia untuk dihantar kepada pengguna sepenuhnya. Dibawah ini adalah aktiviti utama yang dijalankan dalam Model Air Terjun:

- **Analisis.** Mengumpul maklumat keperluan untuk mengenalpasti fungsi dan prestasi perisian..
- **Rekabentuk.sistem.** Keperluan yang dikumpul diterjemahkan ke bentuk perwakilan sistem yang dapat menerangkan bagaimana sesuatu perisian ini dibina.
- **Rekabentuk Program** Rekabentuk dapat membayangkan struktur program, struktur data dan ciri prosedur yang lengkap bagi perisian.
- **Pengkodan.** Rekabentuk mesti diterjemahkan kepada bentuk perlaksana mesin.. Langkah pengkodan dilaksanakan dengan menggunakan bahasa pengaturcaraan konvensional seperti C, C++, Prolog dan lain-lain(bahasa generasi ke-4).
- **Pengujian.** Pengujian adalah aktiviti pelbagai-langkah mengenalpastikan setiap komponen perisian dapat melaksanakan keperluan fungsian Pengujian ini adalah aktiviti pelbagai-langkah yang memastikan setiap komponen perisian dapat berfungsi mengikut keperluan fungsian dan mengesahkan sistem memenuhi semua keperluan pengguna.. Perisian yang telah dibina akan dinilai dan diuji untuk memastikan ia menepati spesifikasi yang telah ditakrifkan. Proses iteratif atau pengulangan akan berlaku untuk memperhaluskan lagi perisian yang telah dibina sehingga ke suatu tahap yang memuaskan pengguna dan pembangun sistem.
- **Penyelenggaraan.** Penyelenggaraan adalah melakukan pembaikan pada setiap langkah aktiviti didalam perisian.Pembaikan ini diperlukan untuk membetulkan segala ralat yang terdapat pada perisian yang dibangunkan dan juga untuk

menyesuaikan perisian dengan perubahan pada persekitaran dalaman (contoh perkakasan baru dan sistem operasi) , atau juga menyediakan peningkatan fungsi atau prestasi oleh pelanggan.

3.2.1.1 Kelebihan dan Kekurangan Model Air Terjun

Kelebihan:

- ❖ Mudah diterangkan kepada pelanggan yang tidak biasa dengan pembangunan perisian
- ❖ Model ini boleh memberikan pembangun perisian pandangan tahap-tinggi semasa proses pembangunan
- ❖ Kebanyakan model lain adalah ubahsuai dari model air terjun
- ❖ Semua R&D dilakukan sebelum pengkodan dimulakan, menghasilkan rekabentuk program yang berkualiti.

Kekurangan:

- ❖ Tidak menggambarkan cara kod dihasilkan – kecuali sesuatu perisian itu sudah benar-benar difahami
- ❖ Tidak menyediakan panduan untuk mengendalikan sebarang perubahan yang berlaku pada produk dan aktiviti
- ❖ Gagal untuk menganggap perisian sebagai satu proses penyelesaian masalah – model air terjun adalah hasil dari proses pembangunan perkakasan
- ❖ Satu fasa perlu disiapkan sebelum ke fasa yang seterusnya.

3.2.2 Model Lingkaran(Spiral)

Model ini menumpukan terhadap pengurangan risiko dalam pembangunan sistem.

Berikut merupakan beberapa risiko yang berkemungkinan untuk dihadapi oleh sistem di sepanjang proses pembangunan sistem mengikut model ini.

1. Sistem mungkin tidak memenuhi keperluan pengguna.
2. Sistem mungkin tidak memenuhi kualiti yang diinginkan.
3. Kos pembangunan sistem melebihi daripada yang diperuntukkan.
4. Masa yang diambil untuk pembangunan sistem melebihi daripada yang diperuntukkan.
5. Pakar-pakar yang terlibat meninggalkan projek sebelum ianya tamat-projek terbengkalai.
6. Pembinaan produk yang sama tetapi lebih bagus daripada projek yang dibangunkan. Ini menyebabkan projek ini ketinggalan dan dianggap usang.

Model ini mementingkan kualiti produk yang dihasilkan disamping berusaha untuk mengurangkan risiko semasa pembangunan sistem. Faktor pengurangan risiko boleh mengurangkan kos dan masa terutamanya apabila tiba fasa pengujian. Analisis risiko yang teliti juga menyebabkan fasa penyelenggaraan menjadi lebih mudah disamping menyediakan beberapa alternatif dalam penyelesaian masalah semasa pembangunan produk atau sistem.

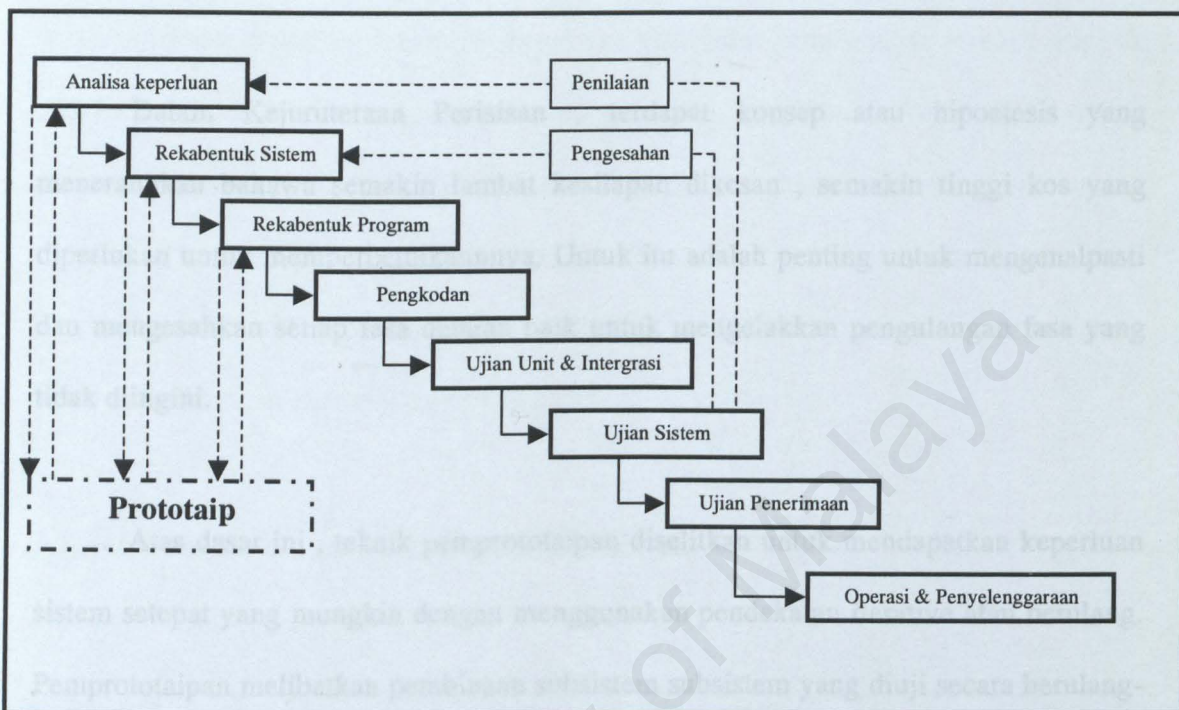
Namun begitu , model ini memerlukan kerjasama yang erat diantara pengguna dan pembangunan sistem untuk mengenalpasti risiko dan kaedah penyelesaiannya. Jadi model ini hanya sesuai untuk pembangunan sistem dalaman sahaja dimana pembangun sistem dan pengguna berada dalam organisasi yang sama. Kerjasama untuk stakeholder dalam organisasi berbeza akan mendatangkan banyak masalah yang mana melibatkan kontrak dan implikasi yang tidak dijangka. Disamping itu , analisis risiko yang tepat bukanlah sesuatu yang mudah untuk dilakukan .Ianya memerlukan masa yang agak lama untuk dipertimbangkan dan diselesaikan. Jadi ianya adalah tidak berbaloi untuk pembangunan sistem yang kecil.

Rajah 3.2 : Model Air Jernih dengan Prototaip

Model Air Jernih (Clean Water Model) oleh Royce pada tahun 1970-an. Model ini memerlukan pembangunan sistem melibatkan analisis keperluan terlebih dahulu. Keperluan sistem ditentukan , dikenalpasti dan disepakati dengan teliti dan dilaksanakan. Selepas spesifikasi ini telah dihasilkan oleh pengguna , barulah fasa seterusnya ditentukan.

Model ini membenarkan maklumbalas dan pengulangan dalam proses pembangunan sistem. Pembangunan sistem boleh berpatah balik kepada fasa yang sebelumnya jika terdapat sebarang perbetulan diperlukan. Sebagai contohnya , semasa

3.2.3 Model Air Terjun dengan Prototaip



Rajah 3.2 : Model Air Terjun dengan Prototaip

Model Air terjun dibina oleh Royce pada tahun 1970 an. Model ini memerlukan pembangun sistem melakukan analisis keperluan terlebih dahulu. Keperluan sistem ditentukan , dikenalpasti dan disemak dengan teliti dan didokumenkan. Selepas spesifikasi ini telah disahkan oleh pengguna , barulah fasa seterusnya ditentukan.

Model ini membenarkan maklumbalas dan pengulangan dalam proses pembangunan sistem. Pembangun sistem boleh berpatah balik kepada fasa yang sebelumnya jika terdapat sebarang pembetulan diperlukan. Sebagai contohnya , semasa

fasa rekabentuk antaramuka pengguna , kesilapan dalam spesifikasinya mungkin memerlukan pembangun sistem berpatah balik kepada fasa analisis keperluan.

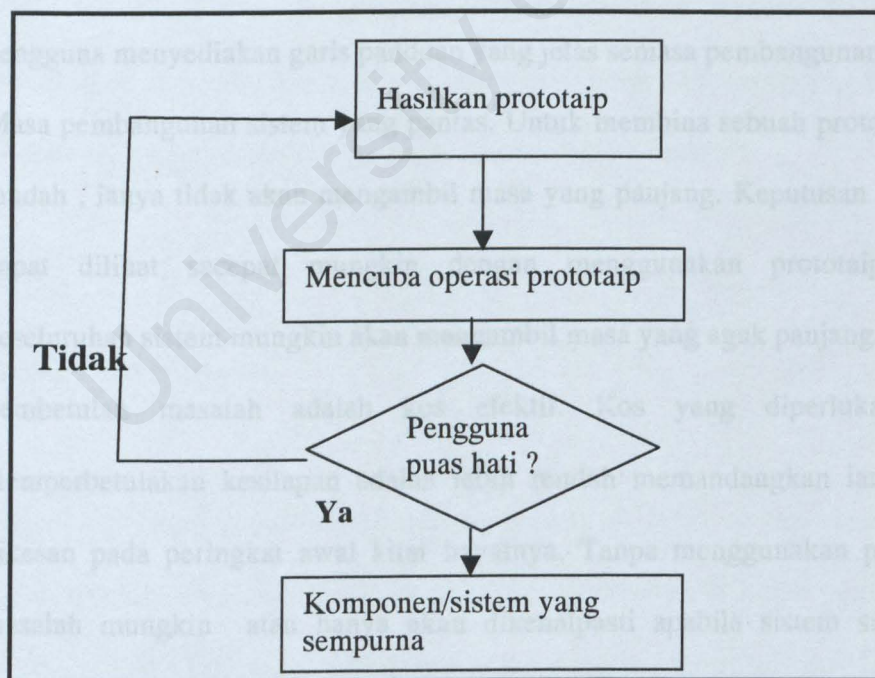
Dalam Kejuruteraan Perisian , terdapat konsep atau hipotesis yang menerangkan bahawa semakin lambat kesilapan dikesan , semakin tinggi kos yang diperlukan untuk memperbetulkannya. Untuk itu adalah penting untuk mengenalpasti dan mengesahkan setiap fasa dengan baik untuk mengelakkan pengulangan fasa yang tidak diingini.

Atas dasar ini , teknik pemprototaipan diselitkan untuk mendapatkan keperluan sistem setepat yang mungkin dengan menggunakan pendekatan iterative atau berulang. Pemprototaipan melibatkan pembinaan subsistem subsistem yang diuji secara berulang-ulang sebelum diimplementasikan sepenuhnya. Dalam membina subsistem -subsistem yang baik dan boleh diterima oleh pengguna , penglibatan pengguna dalam sangat penting. Sekiranya terdapat kesilapan dan pengguna tidak berpuas hati dengan subsistem tersebut , pembangun sistem boleh pergi ke fasa analisis keperluan untuk merekabentuk subsistem yang mengikut spesifikasi yang dikehendaki oleh pengguna. Subsistem yang baru akan diimplementasikan sekali lagi. Proses ini akan dilakukan secara berulang-ulang sehinggalah pengguna benar-benar berpuas hati dengan sistem secara keseluruhannya.

Proses prototaip melibatkan beberapa langkah yang penting. Pembangun sistem akan merekabentuk prototaip dengan pantas .Langkah ini tidak akan memakan masa

yang panjang kerana hanya aspek yang kecil sahaja akan dipertimbangkan. (kerana ia merupakan komponen kecil sistem) .Selepas itu prototaip akan diimplementasikan untuk melihat kebolehlaksanaannya. Pengguna akan turut serta semasa proses ini untuk memberikan tindakbalas terhadap unit-unit prototaip yang dibina. Pembangun sistem akan mengambil kira perkara-perkara yang dicadangkan oleh pengguna untuk melakukan perubahan ke atas unit tersebut. Sebarang perubahan akan melibatkan langkah rekabentuk prototaip dan seterusnya sehinggalah pengguna benar-benar berpuas hati dengan sistem yang dihasilkan.

3.2.3.1 Langkah-Langkah Pemprototaipan



Rajah 3.3 : Perlaksanaan Prototaip

3.2.3.2 Kebaikan Dan Keburukan Prototaip

Kebaikan prototaip

1. Komitmen yang tinggi daripada pengguna dan input yang baik disepanjang proses pembangunan sistem. Salah satu objektif utama dalam pemprototaipan adalah untuk merekabentuk sistem yang memenuhi citarasa dan kehendak pengguna pada peringkat yang paling maksimum. Melalui kaedah ini, pengguna memberikan perhatian terhadap rekabentuk sistem dan komponen-komponen yang membentuknya. Semasa melakukan pemerhatian, pengguna akan memberikan maklumbalas dan maklumbalas terhadap sistem yang diingini. Dengan itu, objektif pembinaan sistem lebih mudah dicapai memandangkan pengguna menyediakan garis panduan yang jelas semasa pembangunan sistem.
2. Masa pembangunan sistem yang pantas. Untuk membina sebuah prototaip yang mudah, ianya tidak akan mengambil masa yang panjang. Keputusan awal juga dapat dilihat secepat mungkin dengan menggunakan prototaip. Tetapi keseluruhan sistem mungkin akan mengambil masa yang agak panjang.
3. Pembetulan masalah adalah kos efektif. Kos yang diperlukan untuk memperbetulkan kesilapan adalah lebih rendah memandangkan ianya boleh dikesan pada peringkat awal kitar hayatnya. Tanpa menggunakan prototaip, masalah mungkin atau hanya akan dikenalpasti apabila sistem sudah siap dibina.

Keburukan Prototaip

1. Pergantungan dan komitmen yang tinggi daripada pengguna. Masalah timbul apabila pengguna tidak mempunyai masa untuk terlibat secara aktif dalam pembangunan sistem. Ini akan mengakibatkan langkah-langkah prototaip terganggu dan ini akan menyebabkan ia tidak dapat disiapkan dalam tempoh yang ditetapkan.
2. Aktiviti-aktiviti menghasilkan prototaip mungkin membawa kepada pembangunan sistem kepada skop yang tidak dirancang dan tidak dijangka. Pengguna sering memerlukan sistem yang baik dan maklumbalas yang diterima mungkin tersasar daripada skop sistem dan menghasilkan sistem yang lebih besar. Ini mungkin menjadikan sistem kurang efektif, lambat disiapkan dan mungkin akan rosak. Pun begitu, pengurusan yang baik sepanjang fasa dalam kitar hayatnya boleh membantu dalam mencegah masalah ini.

3.3 Analisis Dan Sintesis

Metodologi yang dipertimbangkan melibatkan 3 buah model. Seperti yang dibincangkan sebelum ini, Kitar Hayat Pembangunan Perisian, fasa-fasa pembangunan projek ini melibatkan 3 fasa yang penting iaitu analisa sistem dan spesifikasi keperluan, rekabentuk sistem serta implementasi. Oleh itu dalam mempertimbangkan metodologi

yang baik serta sesuai , segala aspek ini perlu diambil kira. Adakah pendekatan yang digunakan oleh model-model ini sesuai untuk pembangunan sistem ini ?

Model pertama , Model Air Terjun , nyata sekali tidak sesuai kerana jika berlaku sebarang pengubahsuaian terhadap keperluan fungsian dan sebarang pembaikpulihan maka ia tidak dapat dilakukan kerana sistem yang dihasil berpandukan model ini tidak mempunyai proses iterasi maka proses pembangunan akan terbengkalai begitu sahaja. Proses pembangunan sistem yang mengambil masa yang terlalu lama untuk mencapai objektif ini. Sistem ini memerlukan interaksi yang berkesan dengan pengguna dan mengambil kira sistem-faktor insani yang menyeluruh dan bersepadu. Penggunaan model ini akan menyukarkan objektif projek dicapai.

Model Lingkaran atau Spiral mempunyai pendekatan yang tersendiri iaitu untuk mengurangkan risiko dalam pembangunan produk atau sistem. Walaupun ianya berkesan dan berguna dalam mengurangkan risiko yang terlibat dalam pembangunan sistem , namun ianya tidak sesuai untuk pembangunan projek yang kecil. Skop projek Pengkategorian Teks adalah kecil dan pengkajian serta penekanan yang mendalam terhadap risiko dianggap tidak sesuai. Faktor masa juga adalah penting dalam projek ini. Analisis risiko yang perlu mengambil kira kaedah penyelesaian yang mengambil masa yang panjang ternyata sekali tidak sesuai untuk projek ini.

Pendekatan model terakhir yang menitikberatkan kepuasan pengguna dan teknik interaktif dianggap sesuai bagi tujuan pembangunan Pengkategorian Teks. Model Air

Terjun dengan Prototaip adalah lebih menjurus kepada sistem-faktor pengguna dan secara tidak langsung turut menyokong aspek-aspek insani dalam pembangunan sistem. Contoh yang paling jelas adalah dari sudut rekabentuk antaramuka , pembangunan sistem bukanlah semata-mata perlu membina antarmuka untuk pengguna , malahan perlu memastikan agar ianya boleh diterima dan disukai oleh pengguna. Bagi mendapatkan antaramuka yang digemari pengguna , pembangun sistem tentunya perlu menjalankan beberapa kajian terhadap Kejuruteraan Insani. Sebab utama model ini dipilih adalah kerana masa pembangunan yang diberi adalah terhad.

3.4 Fokus Terperinci Setiap Fasa Berdasarkan Model Air Terjun

Dengan Prototaip

FASA 1: Analisis Keperluan

Di dalam fasa ini , pengkajian dan analisis terhadap sistem yang sedia ada dijalankan untuk mendapatkan kefahaman yang lebih mendalam terhadap sistem di samping mengenalpasti kelemahan dan kekuatan sistem yang sedia ada. Dengan itu , ianya dapat membantu untuk mencari dan mengenalpasti keperluan tambahan sistem untuk menghasilkan sebuah sistem yang lebih efisien. Terdapat pelbagai bentuk teknik untuk mencapai matlamat ini termasuklah mengkaji dokumen yang sedia ada , mengadakan temubual , menggunakan soalan-soalan dan pelbagai lagi . Fasa ini seharusnya membantu pembangun sistem dari segi pengetahuan terhadap keperluan fungsian dan data disamping mengetahui proses , aliran data dan input-output sesebuah sistem.

Rekabentuk ini meliputi fungsian yang boleh dilaksanakan oleh sistem. Teknik-teknik seperti data Flow Diagram(DFD) , Entity Relationship Diagram(ERD) , carta Struktur dan State Transition Diagram(STD) mungkin boleh digunakan untuk memahami kelakuan sistem secara grafik untuk sistem.

Fasa ini juga perlulah mencapai keperluan pengguna yang dianggap penting untuk pembangunan sistem maklumat seperti antaramuka pengguna(e.g menu , kotak dialog dan tetingkap.) , skrin kemasukan. Semasa fasa ini jugalah , seseorang pembangun sistem perlu mengenalpasti kedua-dua keperluan fungsian dan bukan fungsian. Ini adalah penting dalam memastikan kejayaan pembangunan sesebuah sistem . Kesilapan dalam mengenalpasti keperluan fungsian dan bukan fungsian bukan sahaja menyebabkan sistem ditolak oleh pengguna malahan mengakibatkan kerugian yang besar. Fasa implementasi termasuklah pelaksanaan , pengujian , dokumentasi sistem dan latihan kepada pengguna dan pentadbiran sistem.

FASA 2:Rekabentuk Sistem

Menggunakan maklumat yang diperolehi di dalam fasa analisis , sistem yang dibina perlulah merekabentuk sistem dengan mengambil kira keperluan tambahan dan sebarang perubahan yang diperlukan ke atas sistem yang sedia-ada. Di dalam fasa ini , segala cebisan maklumat yang diperolehi akan disepadukan untuk merekabentuk sistem yang sebenar.

Rekabentuk ini meliputi fungsian yang boleh dilaksanakan oleh sistem termasuklah rekabentuk antaramuka pengguna. Teknik seperti DFD dan ERD yang digunakan dalam fasa pertama boleh juga digunakan untuk fasa rekabentuk sistem untuk mendapatkan kefahaman menyeluruh terhadap rekabentuk sistem sebelum ianya

Rekabentuk sistem menjadi dokumen komunikasi kepada pengguna. Maka ia perlulah mudah untuk mendapatkan maklumbalas yang sepatutnya. Untuk mendapatkan rekabentuk yang berkesan dan diterima pakai oleh pengguna, pendekatan yang akan digunakan adalah pendekatan Human Computer Interactive(HCI)

FASA 3: Implementasi

Fasa implementasi termasuklah pengkodan, pengujian, dokumentasi sistem dan latihan kepada pengguna dan pentadbir sistem.

Pengkodan

Spesifikasi proses ini adalah untuk menghasilkan kod-kod yang menggunakan bahasa pengaturcaraan sistem yang sesuai (e.g Visual C++ ,LISP dll). Untuk kemudahan penyelenggaraan pada masa akan sistem, aturcara yang dibangunkan perlulah mempunyai modul program yang distrukturkan dengan baik.

Pengujian

Pembangun sistem perlu berusaha keras untuk mencapai kecacatan sifar. Tujuan proses ini adalah untuk memperbaiki kesilapan dalam sistem seperti logic dan ketepatan sistem. Pengujian yang baik akan meminimumkan bilangan kesilapan sebelum ianya dihantar kepada pengguna. Proses ini adalah penting untuk mempertingkatkan kualiti sistem.

Pengujian melibatkan beberapa peringkat iaitu:

- 1) Pengujian modul-modul atau unit secara berasingan. Segala kesilapan yang dikenalpasti akan diperbetulkan.
- 2) Pengujian bersepadu yang melibatkan gabungan modul-modul untuk menguji kebolehlaksanaannya.
- 3) Pengujian fungsian untuk memastikan ianya berfungsi sebagaimana yang dikehendaki oleh pengguna.
- 4) Pengujian penerimaan untuk dilakukan oleh pengguna sebelum menerima sistem secara formal.

Penilaian

Penilaian sistem dilakukan selepas sistem beroperasi dalam tempoh 3 hingga 6 bulan. Ini adalah untuk membenarkan pengguna benar-benar mahir menggunakan sistem disamping membenarkan kestabilan sistem. Proses ini penting untuk mengetahui adakah sistem benar-benar mencapai objektif asalnya.

FASA 4: PENYELENGGARAAN

REKABENTUK SISTEM

4.1 Pengenalan

Fasa ini adalah penting untuk memastikan sistem beroperasi secara dinamik dengan persekitarannya. Fasa ini akan dijalankan selepas sistem telah siap dan diserahkan kepada pengguna.

Sebelum ini, penjelasan dan pengetahuan terhadap aspek-aspek penting yang perlu dipertimbangkan dalam pembangunan sesebuah sistem.

Aktiviti analisis sistem memerlukan pendekatan yang berfokus terhadap permasalahan dari sudut pengguna, analisis kerja dan sistem, spesifikasi keperluan yang ditetapkan oleh organisasi tersebut.

Analisis ini adalah penting untuk memastikan sistem melaksana dan menyokong keperluan dan polisi yang sedia ada organisasi tersebut. Untuk tujuan tersebut, pembangunan analisis dibahagikan kepada 2 bahagian yang utama iaitu analisis pengguna dan spesifikasi keperluan sistem yang merangkumi keperluan fungsian dan keperluan bukan fungsian.

4.2 Spesifikasi Keperluan Sistem

Keperluan merupakan ciri-ciri atau deskripsi terhadap apa yang boleh dilakukan oleh sistem untuk memenuhi perancangan sistem yang dicadangkan. Ianya bukan saja menerangkan aliran maklumat keluar dan masuk ke dalam sistem tetapi juga kekangan di dalam pelaksanaan sistem.

BAB 4 : ANALISIS DAN REKABENTUK SISTEM

4.1 Pengenalan

Seperti yang dinyatakan, fasa analisis sistem dan analisis keperluan merupakan fasa terawal dalam kitar hayat pembangunannya. Fasa ini adalah penting untuk mendapatkan penjelasan dan pengetahuan terhadap aspek-aspek penting yang perlu dipertimbangkan dalam pembangunan sesebuah sistem.

Aktiviti analisis sistem memerlukan pendekatan yang terperinci terhadap termasuklah dari sudut pengguna, analisis kerja dan beberapa spesifikasi keperluan yang ditetapkan oleh organisasi tersebut.

Analisis ini adalah penting untuk memastikan sistem melaksana dan menyokong keperluan dan polisi yang sedia ada organisasi tersebut. Untuk tujuan tersebut pembahagian analisis dibahagikan kepada 2 bahagian yang utama iaitu analisis pengguna dan spesifikasi keperluan sistem yang merangkumi keperluan fungsian dan keperluan bukan fungsian.

4.2 Spesifikasi Keperluan Sistem

Keperluan merupakan ciri-ciri atau deskripsi terhadap apa yang boleh dilakukan oleh sistem untuk memenuhi pelaksanaan sistem yang dicadangkan. Ianya bukan saja menerangkan aliran maklumat keluar dan masuk ke dalam sistem tetapi juga kekangan di dalam pelaksanaan sistem.

Untuk mendapatkan keperluan yang tepat memerlukan proses yang berulang dan melibatkan penglibatan yang teguh di antara pengguna dan pembangun sistem. Keperluan yang tepat penting untuk menentukan apakah sistem, manakala rekabentuk menentukan bagaimana pelaksanaan sistem tersebut.

Untuk spesifikasi keperluan sistem yang dibincangkan untuk bahagian ini adalah memfokus kepada keperluan fungsian dan keperluan bukan fungsian.

4.2.1 Keperluan Fungsian

- Pemprosesan indek- beberapa proses dikenalpastikan untuk mengekstrakkan teks dalam dokumen
- Banding perkataan dalam ingatan –teks yang diproses akan dibandingkan dengan ingatan

4.2.2 Keperluan Bukan Fungsian

Beberapa kriteria penting dalam menjamin kepuasan pengguna dalam menggunakan sistem perlu diambilkira. Ianya bukanlah bergantung seratus-peratus pengguna tanpa menghadapi masalah. Keuasan sistem interaktif juga menjadi satu

kepada fungsi-fungsi yang dijalankan oleh sistem , namun turut melibatkan beberapa keperluan bukan fungsian.

Antara keperluan bukan fungsian yang cuba dicapai oleh Pengkategorian Teks termasuklah:

✓ **Kebolehselenggaraan**

Sistem dibangunkan dengan menggunakan pendekatan bermodul yang membahagikan sistem kepada modul-modul kecil. Melalui teknik modulariti ini , setiap komponen mempunyai input , output dan keadaan yang dinyatakan dengan jelas. Ini meningkatkan kefahaman terhadap pelaksanaan sistem dan memudahkan penyelenggaraan pada masa akan sistem.

✓ **Kebolehpercayaan**

Sistem perlu berupaya mengeluarkan output yang dikehendaki oleh pengguna semasa digunakan. Pengesanan ralat dan paparan mesej perlu diimplementasi untuk tujuan ini. Disamping itu , pengujian secara komprehensif perlu dilaksanakan untuk mengesan sebarang kegagalan di sepanjang pembangunan sistem.

✓ **Kecekapan dan Ketepatan**

Sistem perlu mampu memenuhi permintaan pengguna apabila ianya diperlukan walaupun telah digunakan berulang kali. Ianya perlu cekap melayani permintaan pengguna tanpa menghadapi masalah. Kelajuan sistem interaktif juga menjadi satu isu

yang penting dalam aspek penggunaannya. Untuk menjamin kecekapan sistem , beberapa sistem seperti pemprosesan dan simpanan data perlulah dipertimbangkan. Ini adalah kerana operasi pemprosesan mengambil masa yang lebih lama berbanding grafik

✓ **Kebolegunaan**

Sistem perlulah difahami dengan jelas oleh pengguna , mudah digunakan dan diterima oleh pengguna. Sistem juga perlu mampu dipelajari oleh pengguna dalam masa yang singkat walaupun mereka merupakan pengguna yang pertama kali menggunakannya. Dengan itu , sistem perlu mempertimbangkan rekabentuk antaramuka yang bersesuaian dengan pelaksanaan Sistem Pengkategorian Teks ini.

✓ **Boleh difahami**

Merujuk kepada untuk memahami aliran program tersebut. Dengan itu, perubahan boleh dibuat dengan mudah dalam bahagian aturcara yang perlu sahaja tanpa perlu mengubah logik yang lain

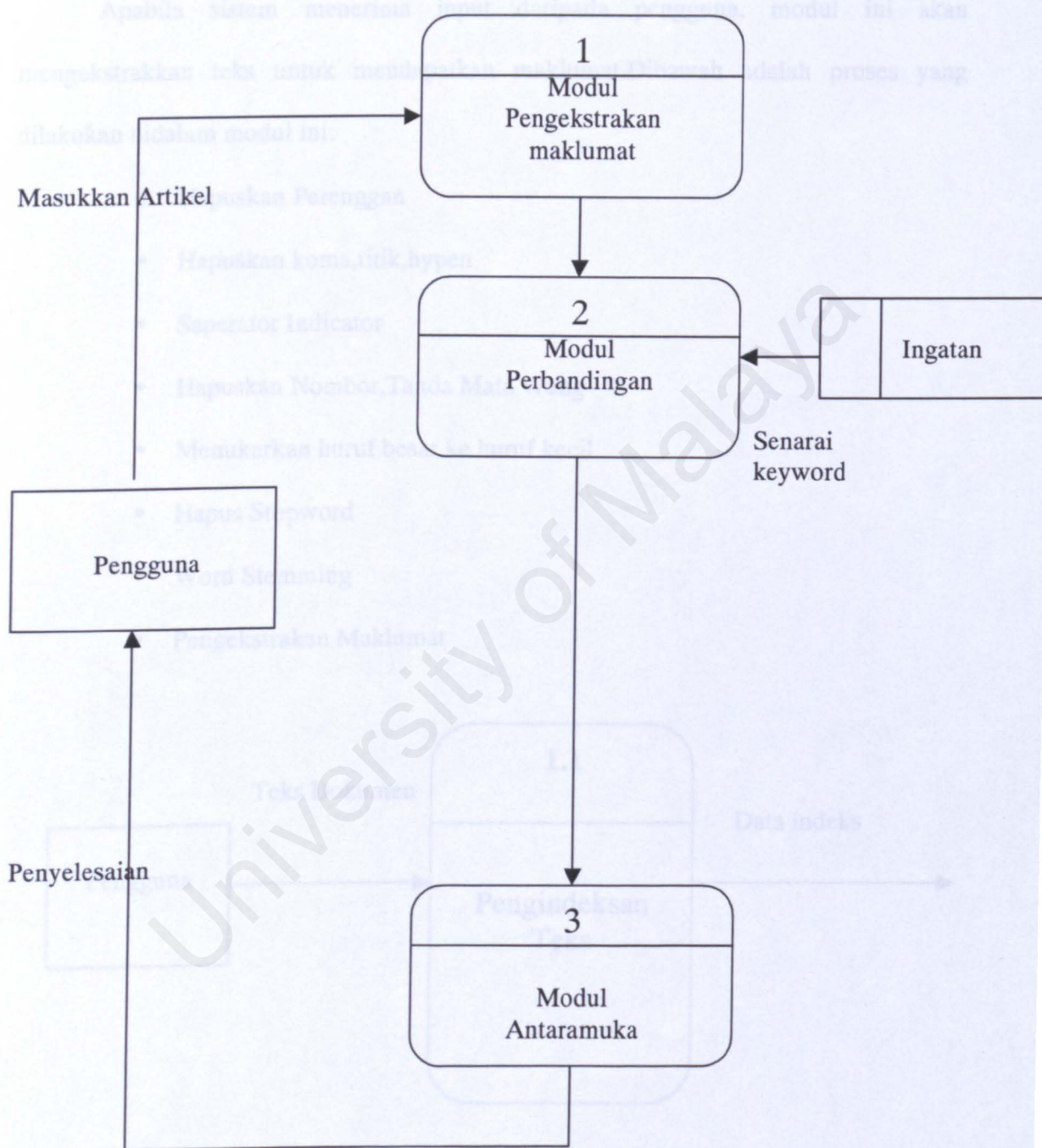
4.3 Rekabentuk Modul

Rekabentuk sistem adalah proses untuk mengubah idea konseptual daripada spesifikasi keperluan dalam analisis sistem kepada spesifikasi yang lebih teknikal. Didalam sistem pengkategorian teks ini terdapat beberapa modul utama yang dikenalpasti iaitu:

- Modul Pengekstrakan Maklumat
- Modul Perbandingan
- Modul Antaramuka

Rajah 4.1 : Carta Alir Sistem

Rajah carta alir data sistem :

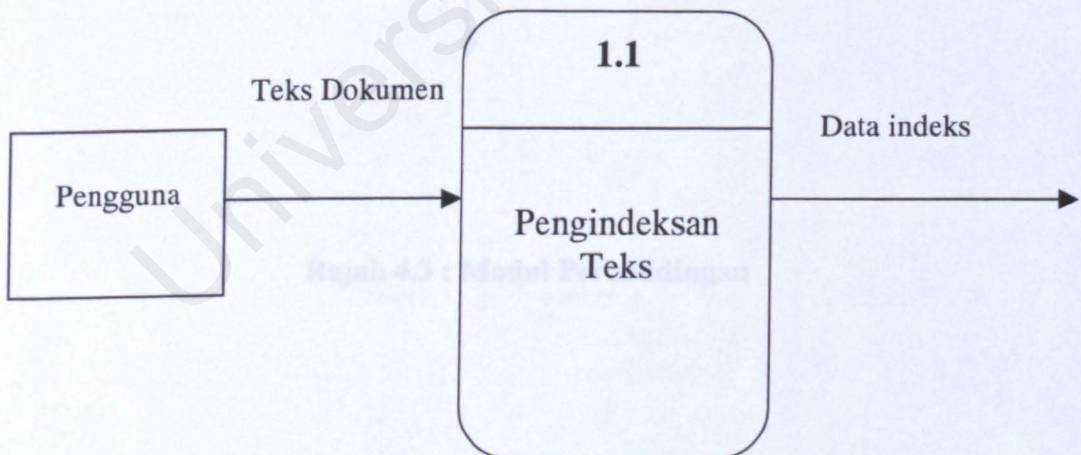


Rajah 4.1 : Carta Alir Sistem

4.3.1 Modul Pengekstrakan Maklumat

Apabila sistem menerima input daripada pengguna, modul ini akan mengekstrakkan teks untuk mendapatkan maklumat. Dibawah adalah proses yang dilakukan didalam modul ini:

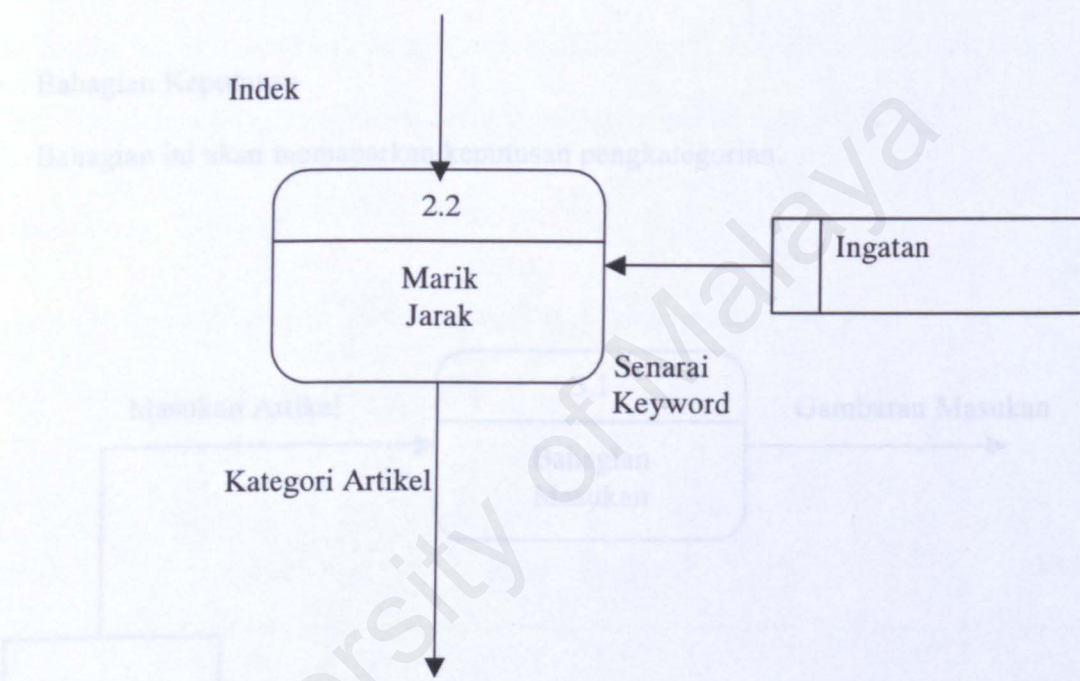
- Hapuskan Perenggan
- Hapuskan koma, titik, hyphen
- Saperator Indicator
- Hapuskan Nombor, Tanda Mata Wang
- Menukarkan huruf besar ke huruf kecil
- Hapus Stopword
- Word Stemming
- Pengekstrakan Maklumat



Rajah 4.2 : Modul Pengekstrakan Maklumat

4.3.2 Modul Perbandingan

Modul ini digunakan untuk membandingkan perkataan yang telah diindek dengan senarai keyword didalam ingatan .Perbandingan dibuat dengan menggunakan matrik jarak



Rajah 4.3 : Modul Perbandingan

4.3.3 Modul Antaramuka

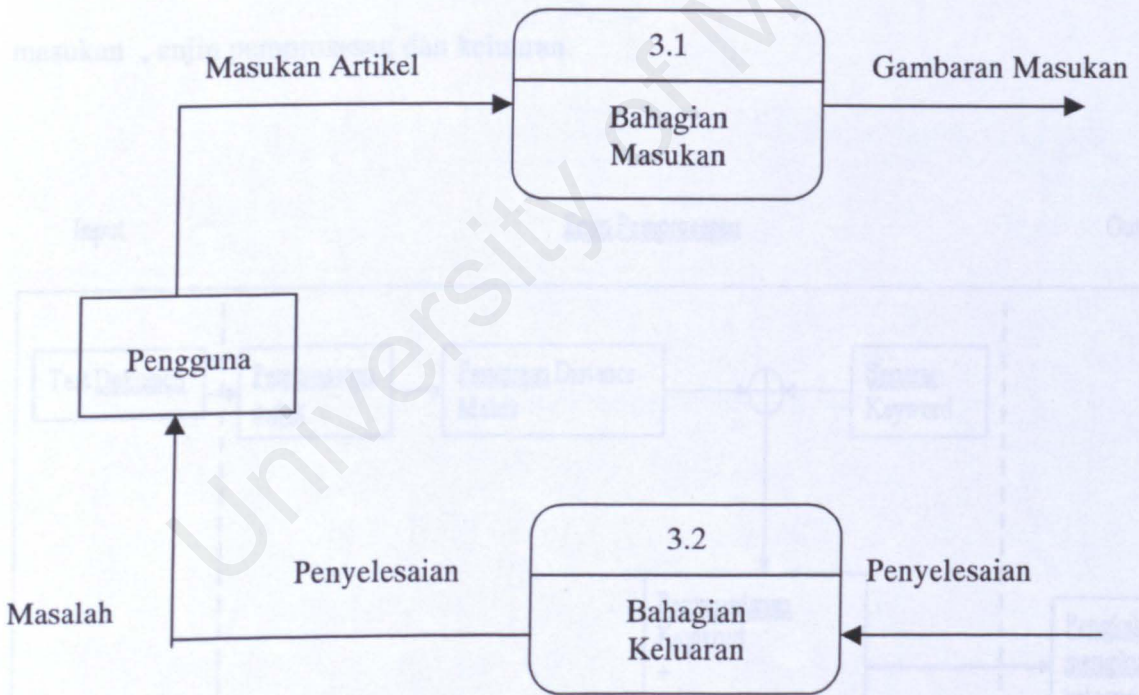
Modul ini mengandungi 2 komponen utama :

- Bahagian Masukan

Pengguna akan memasukkan artikel untuk pengkategorian .

- Bahagian Keputusan

Bahagian ini akan memaparkan keputusan pengkategorian.



Rajah 4.4 : Modul Antaramuka

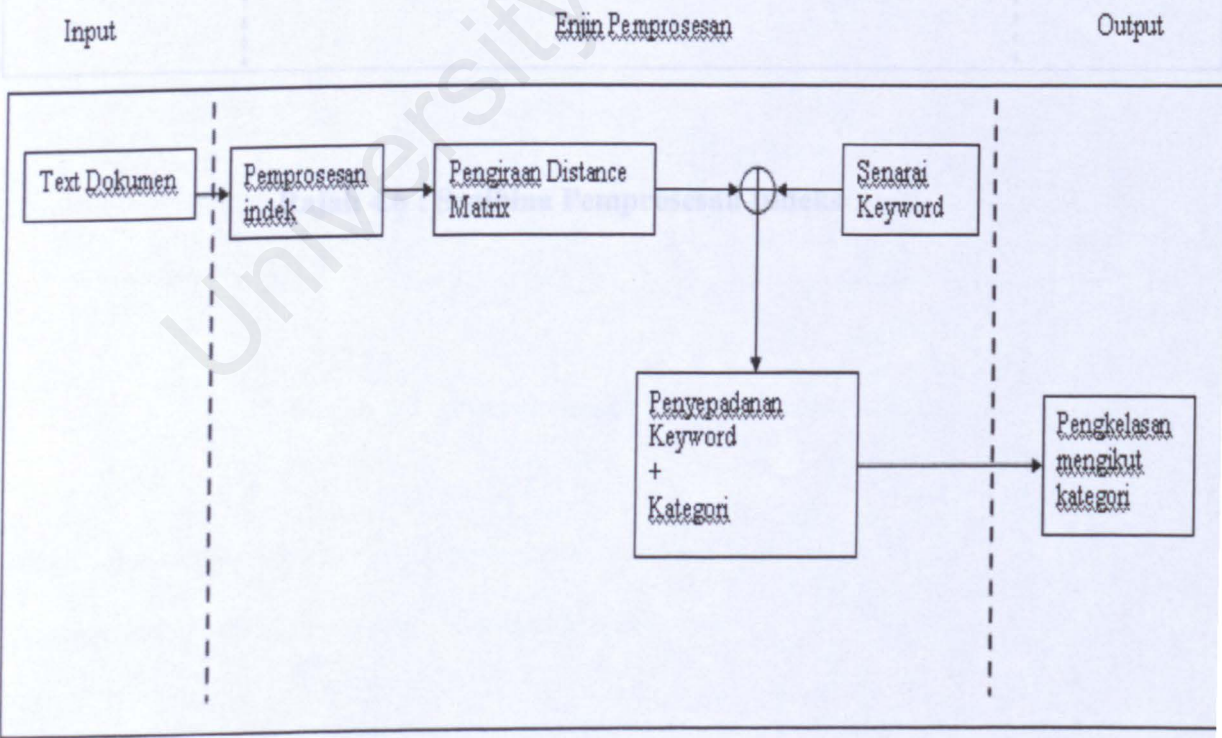
4.4.1 Pemprosesan Indeks

4.4 Rekabentuk Sistem

Setelah mendapatkan keperluan sistem dengan melakukan beberapa analisis , fasa rekabentuk sistem akan dilaksanakan. Fasa ini merupakan peringkat atau fasa dimana segala keperluan sistem akan diterjemahkan kepada karektoristik sistem untuk memenuhi keperluan pengguna dengan cara yang paling mudah untuk difahami.

4.4.1 Senibina Sistem

Sistem Pengkategorian Teks dibahagikan kepada 3 bahagian yang utama iaitu masukan , enjin pemrosesan dan keluaran.

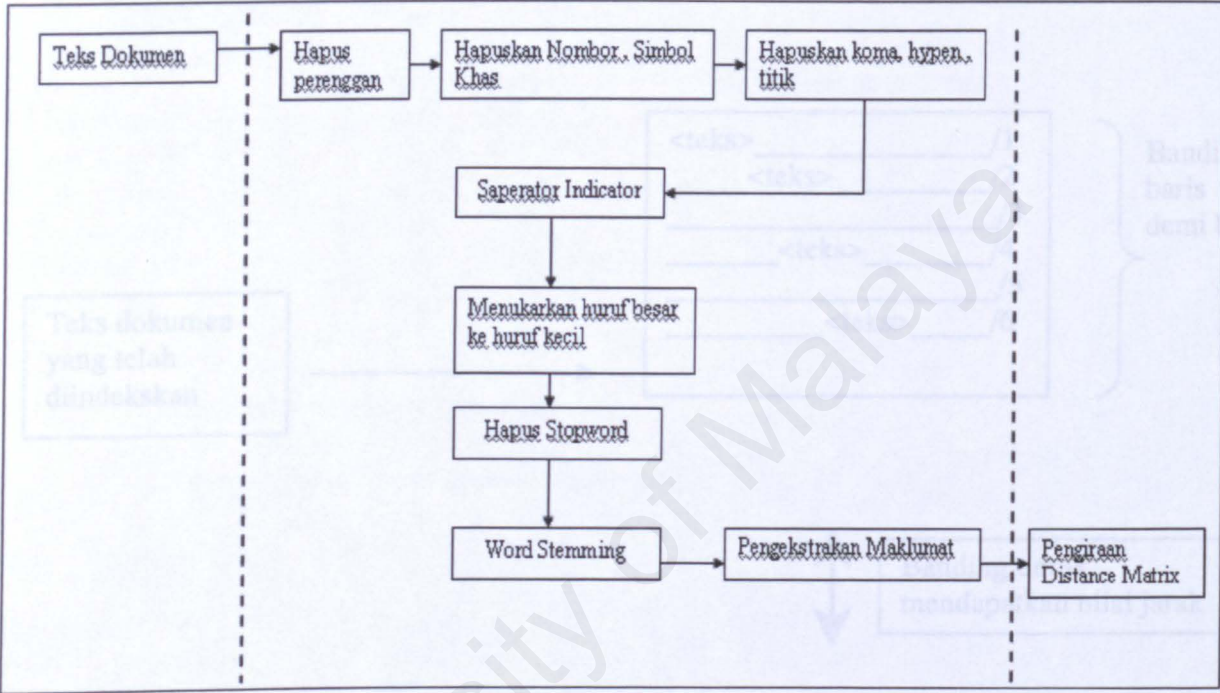


Rajah 4.5 : Senibina Pengkategorian Teks

4.4.1.2 Fungsi Jarak

4.4.1.1 Pemrosesan Indeks

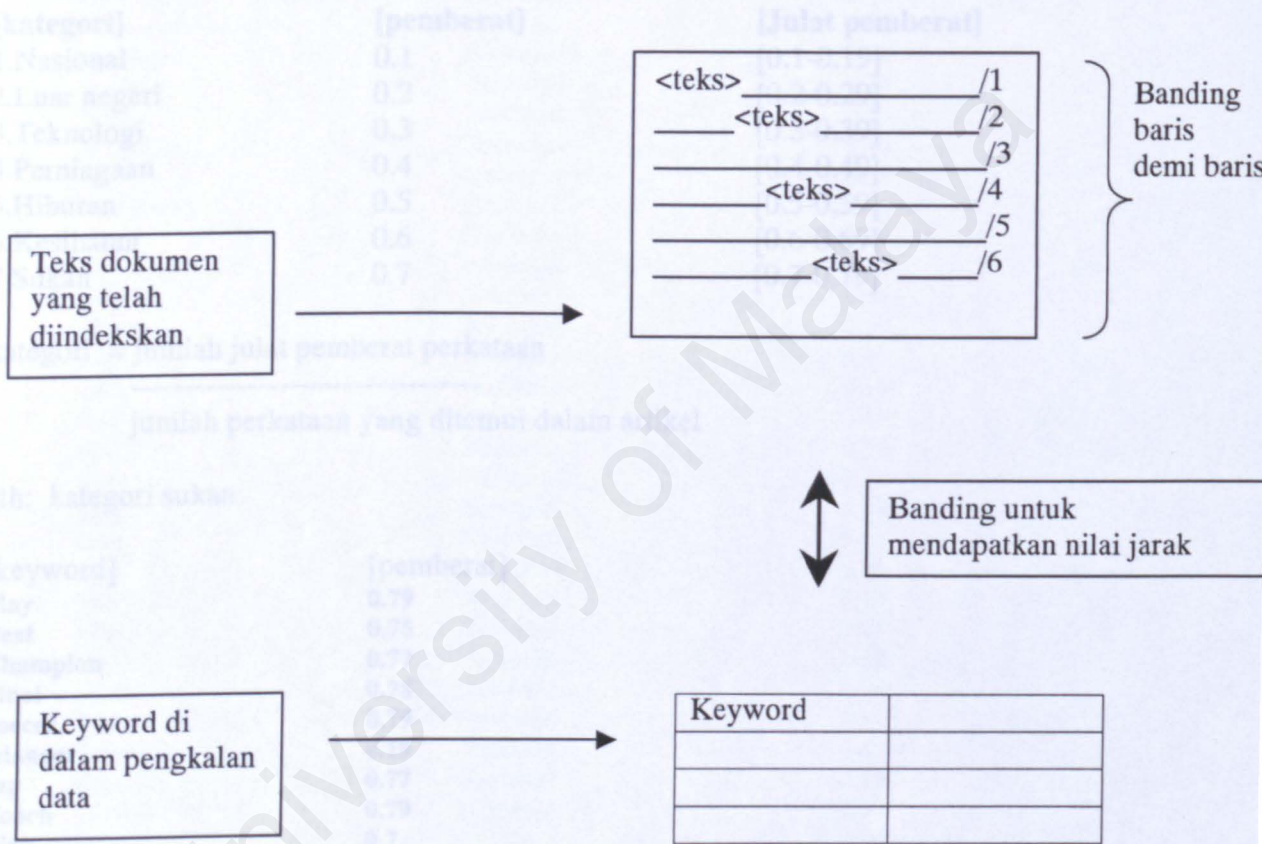
Pemrosesan Indeks



Rajah 4.6 : Senibina Pemrosesan Indeks

4.4.1.2 Fungsi Jarak

Mengira jarak di antara instance yang baru dengan set instance yang ada didalam ingatan untuk menentukan persamaan



Rajah 4.7 : Perlaksanaan Matrik Jarak

Jika terdapat perbezaan kurang daripada <50% dokumen teks berikut akan dikategorikan. Perbandingan dibuat berdasarkan pemberat

4.4.1.3 Keyword

Satu senarai keyword dibina didalam ingatan untuk membandingkan dengan instance baru.Ia akan dikemaskini secara manual untuk memasukkan perkataan yang baru bagi setiap kategori. Terdapat 7 kategori utama iaitu :

[kategori]	[pemberat]	[Julat pemberat]
1.Nasional	0.1	[0.1-0.19]
2.Luar negeri	0.2	[0.2-0.29]
3.Teknologi	0.3	[0.3-0.39]
4.Perniagaan	0.4	[0.4-0.49]
5.Hiburan	0.5	[0.5-0.59]
6.Kesihatan	0.6	[0.6-0.69]
7.Sukan	0.7	[0.7-0.79]

$$\text{kategori} = \frac{\text{jumlah julat pemberat perkataan}}{\text{jumlah perkataan yang ditemui dalam artikel}}$$

cth: kategori sukan

[keyword]	[pemberat]
Play	0.79
Beat	0.75
Champion	0.77
Final	0.78
Soccer	0.79
selangor	0.19
cup	0.77
Coach	0.79
Won	0.7

Jumlah perkataan yang ditemui dalam artikel = 10

Jumlah julat pemberat perkataan = 0.79+0.75+0.77+0.78+0.79+0.7+0.77+0.79+0.7 = 6.84

$$\text{kategori} = \frac{6.84}{10}$$

0.76 dalam kategori **sukan**

Senarai keyword yang dikenalpasti kerap terdapat di dalam sesuatu kategori dan ditetapkan satu pemberat kepada setiap keyword.

[keyword]	[kategori]	[pemberat]
perlis	Nasional	0.19
kedah	Nasional	0.19
perak	Nasional	0.19
pulau pinang	Nasional	0.19
kelantan	Nasional	0.19
terengganu	Nasional	0.19
selangor	Nasional	0.19
kuala lumpur	Nasional	0.19
wilayah persekutuan	Nasional	0.19
negeri sembilan	Nasional	0.19
melaka	Nasional	0.19
pahang	Nasional	0.19
johor	Nasional	0.19
sarawak	Nasional	0.19
sabah	Nasional	0.19
Malaysian	Nasional	0.19
Umno	Nasional	0.19
Pas	Nasional	0.19
Dap	Nasional	0.19
keadilan	Nasional	0.19
kampung	Nasional	0.19
raya	Nasional	0.19
utm	Nasional	0.19
um	Nasional	0.19
uum	Nasional	0.19
upm	Nasional	0.19
usm	Nasional	0.19
uitm	Nasional	0.19
kuittho	Nasional	0.19
suhakam	Nasional	0.19
isa	Nasional	0.19
bukit	Nasional	0.19
Jalan	Nasional	0.19
Taman	Nasional	0.19
Klcc	Nasional	0.19
Proton	Nasional	0.19
Mas	Nasional	0.19
Malay	Nasional	0.19
Malaya	Nasional	0.19
kebangsaan	Nasional	0.19
eon	Nasional	0.19
petronas	Nasional	0.19
mtuc	Nasional	0.19
balai	Nasional	0.19
pprt	Nasional	0.19
tasik	Nasional	0.19
Malaysian	Nasional	0.19
Malaysian	Nasional	0.19
Malaysian	Nasional	0.19
America	Luar Negeri	0.29
war	Luar Negeri	0.2
Singapore	Luar Negeri	0.29
Thailand	Luar Negeri	0.29

Indonesia	Luar Negeri	0.29
Japan	Luar Negeri	0.29
Germany	Luar Negeri	0.29
Iraq	Luar Negeri	0.29
australia	Luar Negeri	0.29
England	Luar Negeri	0.29
Africa	Luar Negeri	0.29
Korea	Luar Negeri	0.29
london	Luar Negeri	0.29
Sains		
robotic	Teknologi	0.39
computer	Teknologi	0.39
electric	Teknologi	0.39
electronic	Teknologi	0.39
mechanical	Teknologi	0.39
machine	Teknologi	0.39
microchip	Teknologi	0.39
processor	Teknologi	0.39
communication	Teknologi	0.39
pc	Teknologi	0.39
network	Teknologi	0.39
hardware	Teknologi	0.39
sortware	Teknologi	0.39
bluetooth	Teknologi	0.39
multimedia	Teknologi	0.39
hardware	Teknologi	0.39
Ekonomi		
system	Perniagaan	0.4
bussiness	Perniagaan	0.49
sale	Perniagaan	0.49
shareholder	Perniagaan	0.49
corporate	Perniagaan	0.49
economy	Perniagaan	0.49
cost	Perniagaan	0.4
account	Perniagaan	0.49
market	Perniagaan	0.49
commertial	Perniagaan	0.4
company	Perniagaan	0.49
plan	Perniagaan	0.4
manage	Perniagaan	0.4
investor	Perniagaan	0.49
consumer	Perniagaan	0.4
product	Perniagaan	0.4
bond	Perniagaan	0.49
manufacture	Perniagaan	0.4
stock	Perniagaan	0.4
Seni		
Film	Hiburan	0.59
Drama	Hiburan	0.59
Opera	Hiburan	0.59
Song	Hiburan	0.59
Entertainment	Hiburan	0.59
Fiction	Hiburan	0.59
Concert	Hiburan	0.59
Festival	Hiburan	0.5

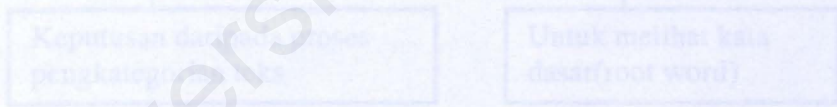
Rock	Hiburan	0.59
Pop	Hiburan	0.59
Movie	Hiburan	0.59
Comedy	Hiburan	0.59
Rap	Hiburan	0.59
Video	Hiburan	0.5
Actress	Hiburan	0.5
Actor	Hiburan	0.5
Compose	Hiburan	0.5
Sing	Hiburan	0.59
Medical	Kesihatan	0.69
Body	Kesihatan	0.6
Treatment	Kesihatan	0.6
Health	Kesihatan	0.69
blood	Kesihatan	0.65
breath	Kesihatan	0.6
ill	Kesihatan	0.65
sore	Kesihatan	0.65
addict	Kesihatan	0.6
heart	Kesihatan	0.66
brain	Kesihatan	0.67
tumor	Kesihatan	0.69
cancer	Kesihatan	0.69
implant	Kesihatan	0.69
doctor	Kesihatan	0.69
pharmacy	Kesihatan	0.69
Play	Sukan	0.7
Game	Sukan	0.7
Beat	Sukan	0.7
Competition	Sukan	0.79
Tournament	Sukan	0.78
Rank	Sukan	0.73
Team	Sukan	0.77
Final	Sukan	0.75
Coach	Sukan	0.77
Rank	Sukan	0.73
Champion	Sukan	0.75
Fight	Sukan	0.75
Won	Sukan	0.75
Lost	Sukan	0.73
Injury	Sukan	0.73
Group	Sukan	0.78
Match	Sukan	0.78
Medal	Sukan	0.77
Challenge	Sukan	0.75
Umpire	Sukan	0.77
referry	Sukan	0.77
league	Sukan	0.77
score	Sukan	0.76
train	Sukan	0.74
opponent	Sukan	0.73
cup	Sukan	0.75

association	Sukan	0.7
round	Sukan	0.7
football	Sukan	0.79
hockey	Sukan	0.79
soccer	Sukan	0.79
gymnastic	Sukan	0.79
baseball	Sukan	0.79
basketball	Sukan	0.79
race	Sukan	0.79
olympic	Sukan	0.79
athletic	Sukan	0.79
golf	Sukan	0.79
criket	Sukan	0.79
snooker	Sukan	0.79
stadium	Sukan	0.7

4.5 Rekabentuk Antaramuka

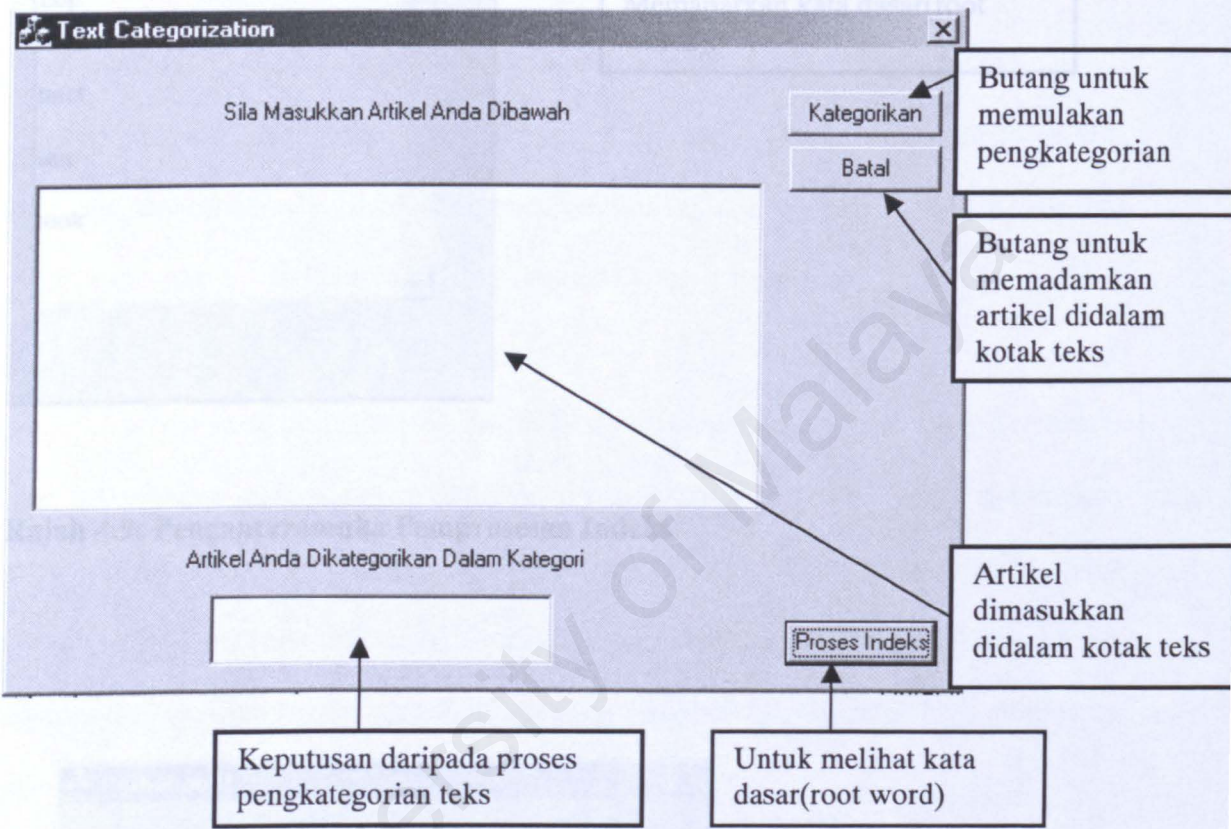
Rekabentuk antaramuka adalah penting dalam fasa rekabentuk sistem. Ianya mempunyai malamat yang agak khusus iaitu menyediakan cara terbaik untuk manusia berkomunikasi dengan komputer. Antaramuka yang baik boleh meningkatkan produktiviti seseorang , kualiti dan perlaksanaan kerja dan keefisyenan sesebuah sistem.

Pengguna memerlukan antaramuka untuk kemudahan menggunakan komputer atau disamping mnembantu mereka dalam menggunakan sistem tersebut dengan lebih mudah dan cepat. Pengguna biasa tidak berminat untuk mengetahui teknologi di sebalik komputer dan juga mengambil masa yang banyak untuk mempelajari perisian komputer tersebut. Mereka memerlukan komputer untuk melaksanakan kerja dengan lebih mudah. Antaramuka yang baik pastinya dapat membantu dalam mencapai objektif ini.

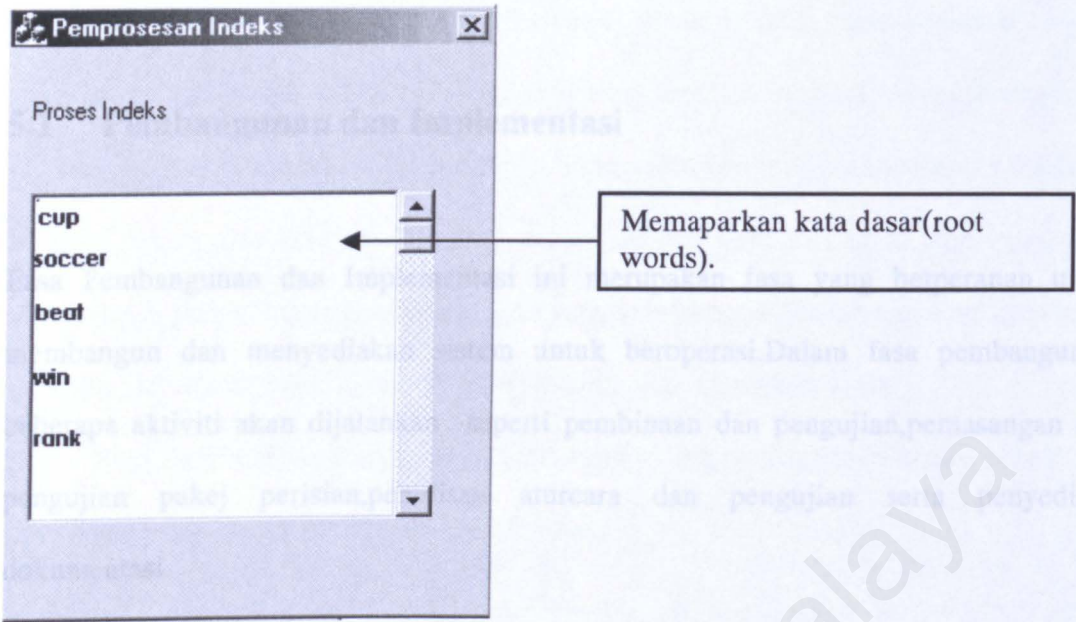


Rekabentuk 4.8 : Pengantaramuka sistem yang direka

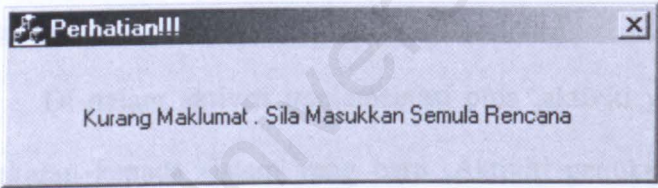
Contoh Rekabentuk yang dicadangkan .



Rajah 4.8 : Pengantaramuka sistem yang dicadangkan



Rajah 4.9: Pengantaramuka Pemrosesan Indeks



Rajah 4.10 Pengantaramuka Amaran

BAB 5 : IMPLEMENTASI

5.1 Pembangunan dan Implementasi

Fasa Pembangunan dan Implementasi ini merupakan fasa yang berperanan untuk membangun dan menyediakan sistem untuk beroperasi. Dalam fasa pembangunan, beberapa aktiviti akan dijalankan seperti pembinaan dan pengujian, pemasangan dan pengujian pakej perisian, penulisan aturcara dan pengujian serta penyediaan dokumentasi.

Aktiviti yang paling utama dalam fasa pembangunan ini ialah pengaturcaraan. Menulis aturcara merupakan aktiviti yang paling penting kerana ianya merupakan aktiviti yang akan merealisasikan segala perancangan yang dibuat pada semester yang lepas. Aktiviti menulis ini telah mengambil masa yang agak singkat kerana telah dibantu oleh perisian pembangunan sistem yang menyediakan “wizard” bagi membantu pengaturcaraan.

Di dalam aktiviti implementasi pula, aktiviti yang paling utama ialah aktiviti penukaran kepada sistem yang baru. Aktiviti penukaran boleh dikatakan kemuncak kepada fasa implementasi. Ini kerana, aktiviti tersebut akan menunjukkan samada sistem yang dibangunkan itu mampu beroperasi seperti yang dijangkakan ataupun tidak. Selain daripada itu, pengujian sistem dijalankan dalam fasa ini.

Pembangunan sistem boleh didefinisikan sebagai satu proses untuk membangun, memasang dan menguji komponen-komponen sistem yang dibangunkan. Manakala fasa implementasi pula boleh didefinisikan sebagai penyerahan sistem yang dibangunkan itu untuk beroperasi.

Kedua-dua fasa ini boleh dikatakan sebagai fasa merealisasikan segala perancangan yang telah dibuat pada fasa-fasa yang sebelum ini. Bagi fasa pembangunan, jangka masa perjalanan fasa ini boleh dipendekkan jika kesemua perancangan yang telah dibuat di dalam fasa-fasa sebelumnya dibuat dengan teliti. Jika tidak, apa yang akan berlaku ialah pengulangan proses dari fasa yang sebelumnya disebabkan kesilapan yang perlu dibaiki di fasa tersebut, dan ini sudah tentu akan memakan masa.

Bagi fasa implementasi pula, aktiviti yang paling penting ialah proses penukaran dari sistem yang lama kepada sistem yang baru.

5.2 Objektif fasa pembangunan dan implementasi

Objektif utama bagi fasa pembangunan ialah bagi memastikan sistem yang akan dibangunkan nanti mengikut segala perancangan yang telah dibuat di dalam fasa-fasa yang sebelumnya. Fasa ini berperanan untuk menguji kefungsi sistem tersebut bagi memastikan ianya memenuhi keperluan-keperluan dan rekabentuk sistem yang sebenar. Di samping itu, ia juga bertanggungjawab untuk memastikan antaramuka bagi sistem yang baru dan yang lama berfungsi dengan baik.

5.2.1 Fasa pembangunan

Pengaturcaraan merupakan aktiviti utama dalam fasa ini. Disamping itu juga, ianya digunakan di dalam proses pembangunan sistem yang baru ia juga diperlukan untuk tujuan mengimplimentasi dan intergrasi komponen-komponen sistem yang baru dengan yang lama.

Penyediaan sistem dibangunkan oleh MFC. MFC adalah Microsoft Foundation Class Library (MFC) , menyediakan bingkai (framework) MFC dan komponen dan subsistemnya. Kelas di dalam Microsoft Foundation Class Library (MFC) akan membuat satu "bingkai aplikasi" — bingkai di mana ia dibina untuk aplikasi Window. Pada setiap peringkat umum bingkai akan menyediakan kerangka aplikasi utama memberikan implementasi antaramuka pengguna yang boleh diletakkan di kerangka. Pembangun hanya memasukkan data-data atau fungsi-fungsi yang dikehendaki ke dalam kerangka-di mana ia akan menspesifikasikan aplikasi sistem. Dengan menggunakan AppWizard, pembangun boleh mencipta fail pada pemula aplikasi. Pembangun menggunakan sumber penyunting Microsoft Visual C++[™] untuk merekabentuk elemen antaramuka pengguna secara visual, ClassWizard akan menyambungkan elemen ke kod dan perpustakaan kelas untuk mengimplementasikan aplikasi logik sistem.

Di dalam MFC akan menyediakan Dynamic link libraries (DLLs) iaitu satu ciri yang amat berguna di dalam sistem. Pada masa sekarang didapati pembangun akan menggunakan pelbagai penyambung DLL. Ia akan membenarkan modul kod yang boleh laksana dimasukkan pada penyambung dan peminta pada masa lari. Ini

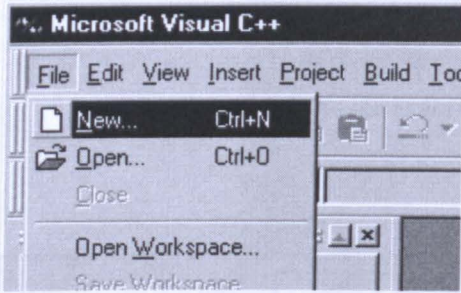
membolehkan kod-perpustakaan mengemaskini secara automatik, transparent kepada aplikasi, dan kemudian memutuskan bila tidak diperlukan.

Terdapat dua kebaikan utama menggunakan penyambung masa-lari(run-time):

- Apabila DLL tiada , aplikasi menggunakan penyambung dinamik load-time akan dihapuskan, manakala penyambung dinamik masa-lari(run-time) boleh bertindakbalas kepada ralat.
- Apabila DLL berubah , aplikasi yang menggunakan penyambung dinamik load-time akan menghapus semula , sementara aplikasi penyambung dinamik masa-lari(run-time) akan dikesan jika fungsi yang diperlukan tiada didalam DLL baru.

5.3 Awal pembangunan

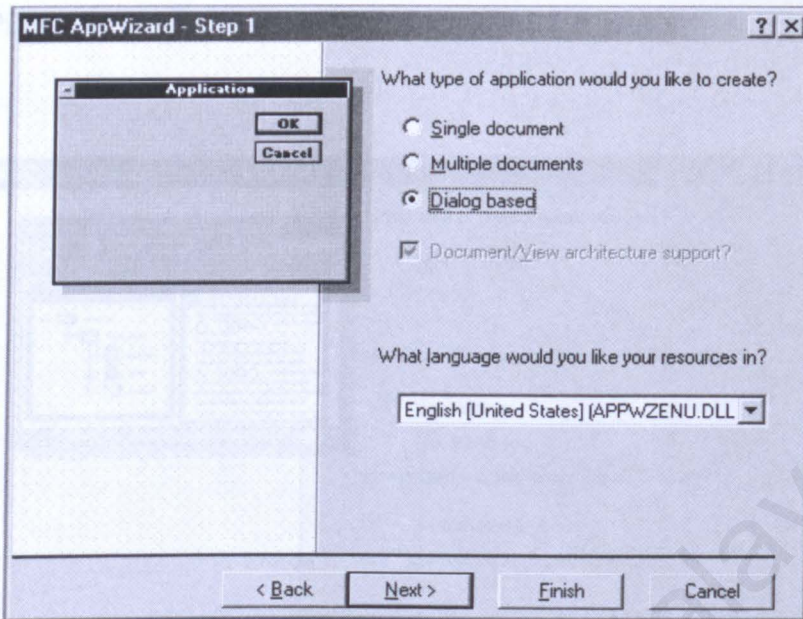
- Lancarkan persekitaran Microsoft Visual C++ .
- Pilih File and New...



Rajah 5.1 : Buka File

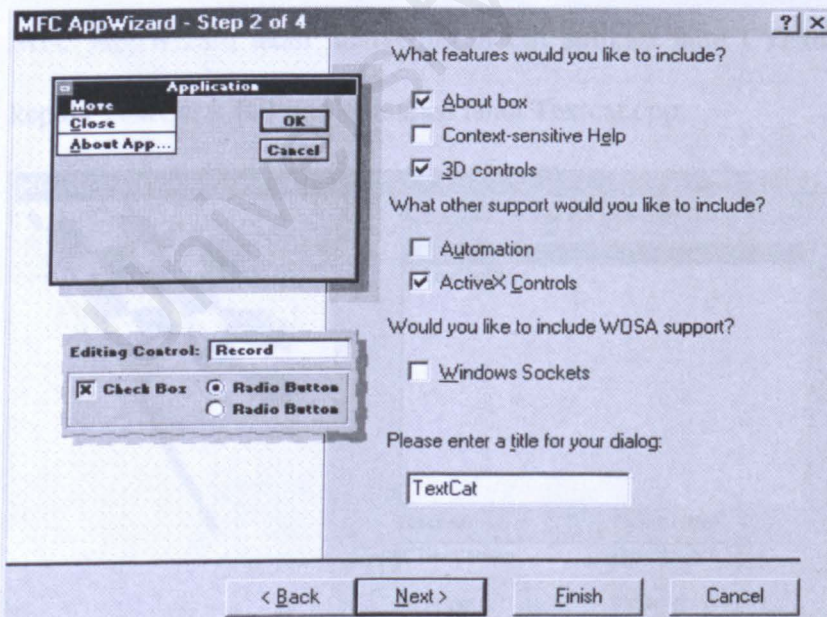
- Masukkan nama item di dalam kotak About > JD controls,ActiveX Control.

- Pada pilihan New dialog box, klik Projects property sheet dan klik MFC AppWizard (exe)
- Dalam kotak Project Name, taipkan TextCat , tandakan pada platform win32 dan klik OK.
- Klik pada Dialog Based->Next



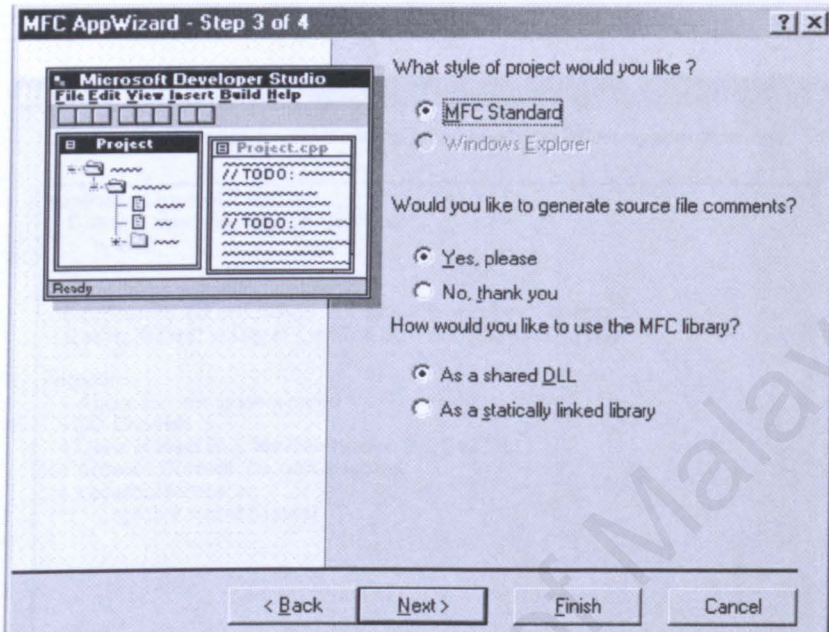
Rajah 5.2 : MFC Langkah 1

- Masukkan nama untuk dialog,pilih About box,3D controls,ActiveX Control.



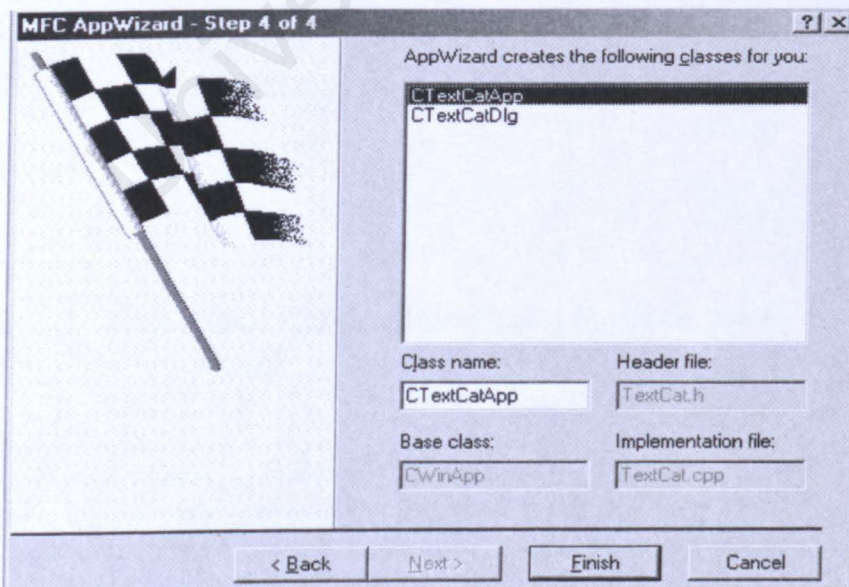
Rajah 5.3 : MFC Langkah 2

- Pilih pilihan MFC standard dan shared DLL



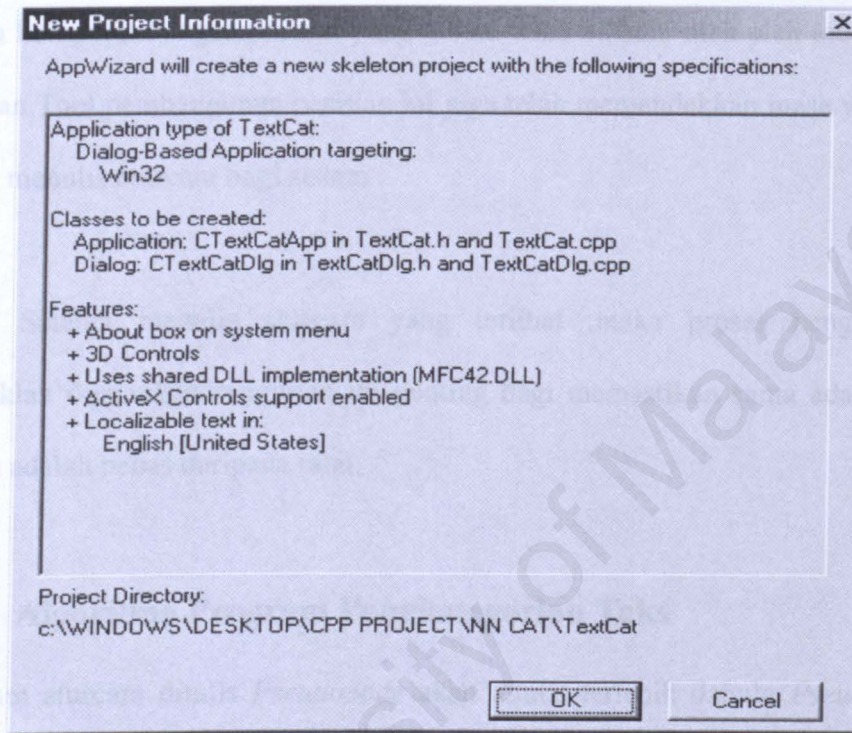
Rajah 5.4 : MFC Langkah 3

- MFC AppWizard akan menyediakan fail aplikasi iaitu CTextCatApp dan fail kepala TextCat.h, fail implementasi ialah Textcat.cpp.



Rajah 5.5 : MFC Langkah 4

5. • Dan akhir sekali MFC AppWizard akan menghasilkan satu laporan mengenai cirri-ciri dan fail yang telah dibina



Rajah 5.6 : AppWizard Menghasilkan Rangka Projek

5.4 Penulisan aturcara dan pengujian

Aktiviti penulisan dan pengujian aturcara ini merupakan satu aktiviti yang akan merealisasikan segala perancangan yang dibuat di dalam fasa-fasa yang terdahulu ke dalam komputer. Pengaturcaraan yang dibuat telah dibantu oleh tool pembangunan perisian. Tool pembangunan perisian ini juga telah memendekkan masa yang diperlukan untuk menulis aturcara bagi sistem.

Setelah menulis aturcara yang terlibat, maka proses pengujian aturcara hendaklah dijalankan. Pengujian ini penting bagi memastikan sama ada aturcara yang dibina adalah bebas daripada ralat.

5.4.1 Algoritma Program Pengkategorian Teks

Sebelum aturcara ditulis *Pseudocode* akan ditulis terlebih dahulu. *Pseudocode* adalah bahasa maklumat yang dapat membantu pengaturcara membangunkan algoritma. *Pseudocode* yang dibangunkan ini amat berguna untuk membina algoritma di mana ia akan ditukarkan kepada struktur program C++. Algoritma yang ditulis adalah seperti berikut :

5.4.1.1 Algoritma untuk dialog memasukkan teks

Masukkan teks(string)

Setkan panjang perkataan yang boleh diterima

Dapatkan panjang perkataan melalui fungsi GetLength()

Jika panjang perkataan besar daripada jumlah yang boleh diterima

Tukarkan semua huruf besar ke kecil

Setkan posisi =0

Count=0

Senaraikan nombor,symbol khas, titik dan koma yang hendak dihapuskan

while panjang perkataan kurang daripada 15

while no kurang daripada 42 senarai

while posisi kurang daripada panjang perkataan

cari nombor,symbol khas, titik dan koma yang hendak dihapuskan

jika jumpa hapuskan

posisi tambah 1

count tambah 1

n > daripada 15 akan cari aksara yang tinggal

updatedata

setkan posisi kepada 0

keluarkan masej(bagi memudahkan untuk melihat kejayaan fungsi)

simpan teks dalam pembolehubah sementara

setkan jumlahperkataan =0

setkan jumlahpemberat=0

buka fail keyword txt untuk baca *Jika* fail sama dengan kosong

Masej tidak boleh buka fail

Else

For

Set Teks yang dimasukkan ke satu pembolehubah Sementara(temp).

jika

temp == -1

else

bandingkan keyword dengan teks

while fail tidak eof

delete format dalam fail keyword

delete (,0 dan .)

setkan val=0

jika jumlah perkataan tidak sama = 0

val= jumlah score bahagi dengan jumlah perkataan keyword

jika val tidak sama = 0

buka fail category txt untuk baca *Jika* fail sama dengan kosong

Masej tidak boleh buka fail

While fail tidak eof

Bandingkan hasil jumlah bahagi dengan julat kategori

Jika tidak jumpa keluarkan masej tiada dalam kategori

Jika terdapat dalam julat keluarkan masej dalam kategori tertentu

Else

Keluarkan masej kurang jumlah perkataan untuk diproses

5.4.2 Pengujian Modul Dan Unit

5.4.1.2 Algoritma untuk dialog memasukkan keyword dalam pengkalan data

Setkan Temp sebagai CString

buka fail keyword.txt untuk baca dan tulis *Jika* fail sama dengan kosong

Masej tidak boleh buka fail

Else

Set format keyword, kategori, pemberat yang hendak dimasukkan ke dalam fail

Letakkan keyword, kategori, pemberat ke dalam fail keyword

Updatedata()

5.4.2 Pengujian Modul Dan Unit

Pengujian modul ini merujuk kepada pengujian kefungsian bagi setiap modul secara individu. Pengujian ini bertujuan mengesan pepijat diperingkat yang paling awal iaitu diperingkat modul. Pengujian ini dilakukan dengan teratur dan teliti, bagi memastikan sistem yang dibangunkan tidak menghadapi sebarang masalah besar apabila diuji pada peringkat yang seterusnya.

Pengujian unit ini pula merujuk kepada pengujian kefungsian intergrasi bagi modul-modul yang dibina. Walaupun setiap modul itu berjaya di dalam peringkat modul, tidak bermakna yang ianya akan berjaya di dalam ujian diperingkat ini. Di dalam pengujian unit ini, kebolehan modul-modul untuk berinteraksi antara satu sama lain diuji.

Rajah 5.7 : Rajah masej untuk memverifikasi fungsi

Arahan ini akan diletakkan pada setiap fungsi yang utama di dalam aliran. Setiap kesalahan atau ketidakfungsian sesuatu arahan dapat dikesan dan dirumuskan.

5.4.2.1 Pengujian sistem

Dengan menggunakan arahan

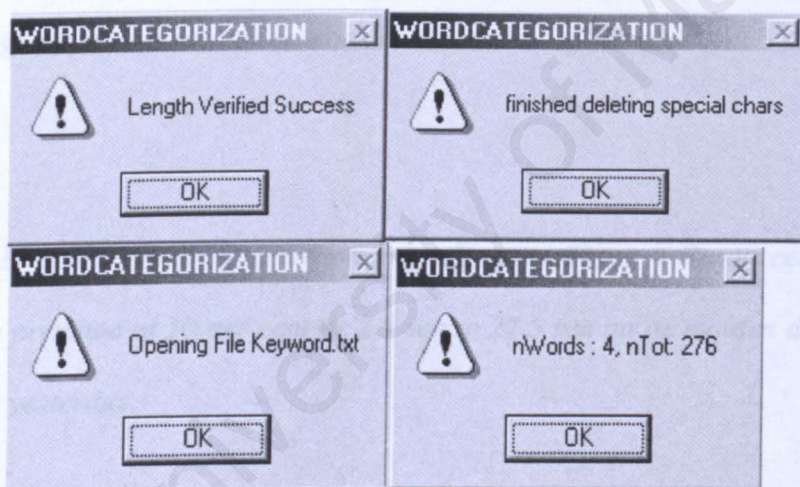
```
AfxMessageBox("Length Verified Success ");
```

```
AfxMessageBox("finished deleting special chars");
```

```
AfxMessageBox("Key Word.txt open success");
```

```
AfxMessageBox(csTemp2);
```

```
AfxMessageBox("Cannot Open file Category.txt");
```



Rajah 5.7 : Rajah masej untuk memeriksa fungsi

Arahan ini akan diletakkan pada setiap fungsi yang utama di dalam aturcara. Setiap kesalahan atau ketidakfungsian sesuatu arahan dapat dikesan dan ditunjukkan.

5.4.2.1 Pengujian sistem

Setelah tamat pengujian unit ,maka barulah keseluruhan modul-modul itu akan diuji sebagai sistem yang lengkap.Pengujian sistem ini melibatkan pengujian sistem maklumat yang dibina dengan menggunakan data-data yang sebenar di dalam suasana yang sebenar .Pengujian ini adalah untuk memastikan bahawa sistem tersebut telah benar-benar bersedia untuk digunakan.

Antara pengujian data sebenar yang telah dilakukan ialah dengan memasukan satu perenggan ayat dari keratan rencana ke dalam dialog antaramuka :

1.FOOD and beverage ingredients manufacturer Three-A Resources Bhd recorded a closing premium of 10 per cent or 2.5 sen to 27.5 sen on its maiden day on the Mesdaq Market yesterday.

2. Indra, 21, voted the most promising footballer by the Perak FA, plays in a similar position as Syed Ahmad did in the 70s. The lanky player, given an attacking midfielder's role by former coach Karl Weigang, has been transformed into a top striker by present coach Tony Netto and will fill in for Khalid Jamlus, who is in Germany with a Division Two club.

3. Three-A manufactures over 17 different varieties of caramel colour, HVP, natural fermented vinegar (NFV) and glucose/maltose syrup and exports about 30 per cent of these products to 13 countries which include Hong Kong, Taiwan, China, Indonesia, Myanmar, the Philippines, Singapore, the US, Australia, Poland, Vietnam, Cambodia and Thailand.

menggunakan sistem tersebut.

Penyediaan dokumentasi merupakan aktivitas yang tidak kurang pentingnya di dalam fase pembangunan ini. Tanpa dokumentasi yang baik dan lengkap, sebuah sistem ini akan menghadapi masalah di dalam operasi penyediaan sistemnya.

Di dalam laporan ini akan menyediakan dua jenis dokumentasi yaitu dokumentasi pengguna dan dokumentasi teknikal.

5.4.3.1 Dokumentasi Teknis

Dokumentasi Teknis akan memberikan setiap fungsi-fungsi di dalam sistem ini kepada pengguna untuk dialog yang dibina.

Terdapat 3 dialog utama di dalam sistem ini iaitu:

- Dialog About
- Dialog Keyword Data Entry
- Dialog Text Categorization

5.4.3 Penyediaan Dokumentasi

Penyediaan dokumentasi merupakan aktiviti yang akan merekodkan segala fakta dan spesifikasi bagi sesebuah sistem untuk tujuan rujukan masa sekarang dan masa hadapan. Rujukan yang dimaksudkan boleh jadi untuk memahami sistem tersebut dengan lebih dekat untuk tujuan penyelenggaraan atau untuk membuat perubahan tanpa perlu menggunakan sistem tersebut.

Penyediaan dokumentasi merupakan aktiviti yang tidak kurang pentingnya di dalam fasa pembangunan ini. Tanpa dokumentasi yang baik dan lengkap, sesebuah sistem itu akan menghadapi masalah di dalam operasi penyelenggaraannya.

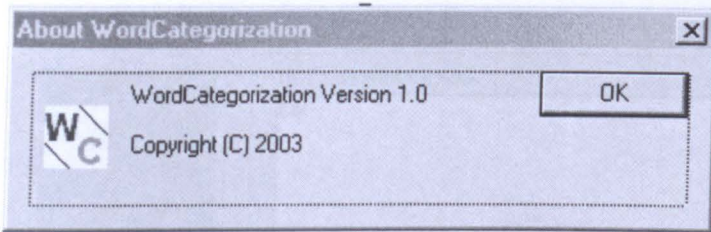
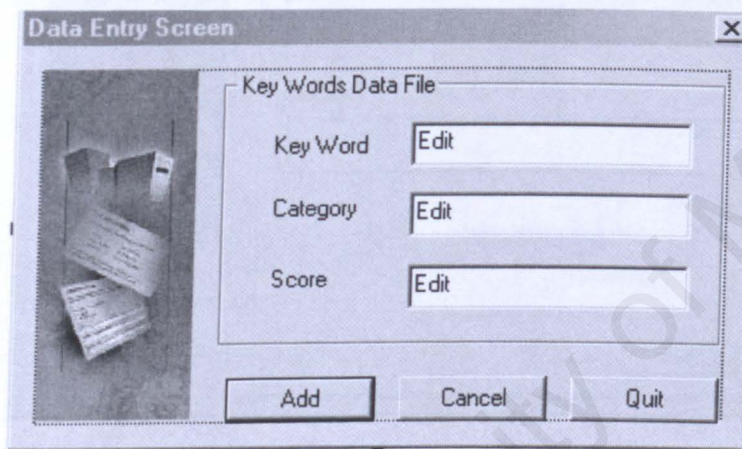
Di dalam laporan ini akan menyediakan dua jenis dokumentasi iaitu dokumentasi pengguna dan dokumentasi teknikal.

5.4.3.1 Dokumentasi Teknikal

Dokumentasi Teknikal akan menerangkan setiap fungsi-fungsi di dalam aturcara, pembolehubah dan dialog yang dibina.

Terdapat 3 dialog utama di dalam sistem ini iaitu:

- Dialog About
- Dialog Keyword Data Entry
- Dialog Text Categorization

Dialog About**Rajah 5.8 : Dialog About****Dialog Keyword Data Entry****Rajah 5.9 : Dialog Keyword Data Entry**

Dialog Data Entry Screen diwakili oleh `IDD_KEYWORDDLG_DIALOG`

Kotak edit bagi keyword diwakili oleh `IDC_KEYWORD`

Kotak edit bagi category diwakili oleh `IDC_CATEGORY`

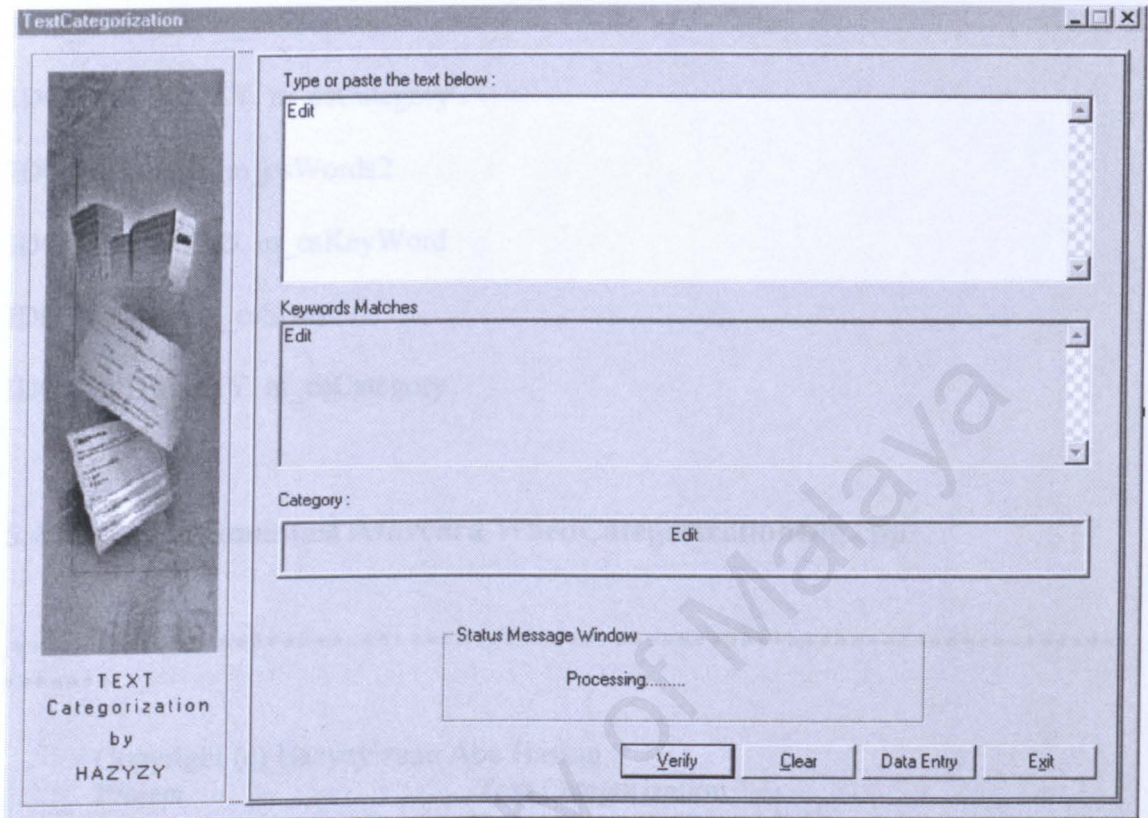
Kotak edit bagi score diwakili oleh `IDC_SCORE`

Butang Add diwakili oleh `IDADD`

Butang Cancel diwakili oleh `IDCANCEL`

Butang Quit diwakili oleh `IDQUIT`

Dialog Text Categorization



Rajah 5.10 : Dialog Text Categorization

Dialog Text Catedorization diwakili oleh :

IDD_TEXTCATEGORIZATION_DIALOG

Kotak edit bagi Type or Paste Below diwakili oleh IDC_WORDS

Kotak edit bagi keywords matches diwakili oleh IDC_WORDS2

Kotak edit bagi category diwakili oleh IDC_CATEGORY

Teks Processing diwakili IDC_STATIC_STATUS

Butang Verify diwakili oleh IDVERIFY

Butang Clear diwakili oleh IDC_CLEAR

Butang Data Entry diwakili oleh IDC_DATAENTRY

Butang Exit diwakili oleh IDEXIT

Data-data yang digunakan

IDC_WORDS, m_csWords

IDC_CATEGORY, m_csCategory

IDC_WORDS2, m_csWords2

IDC_KEYWORD, m_csKeyWord

IDC_SCORE, m_csScore

IDC_CATEGORY, m_csCategory

5.4.3.2.1 Dokumentasi Aturcara WordCategorizationDlg.cpp

```

/*****
*****/

```

```

// Implementation of WordCategorizationDlg.cpp
Copyright (c) Hazyzyizuan Abu Hassan

```

```

Project      : Text Categorization

```

```

FileName     : WordCategorizationDlg.cpp

```

```

Comments     : Implementation file -to categorize text into
               specify category

```

```

Written by   : Hazyzyizuan Abu Hassan
              hazyzy77@yahoo.com

```

```

*****/

```

```

#include "stdafx.h"

```

```

#include "WordCategorization.h"

```

```

#include "WordCategorizationDlg.h"

```

```

#include "KeyWordDlg.h"

```

```

#ifdef _DEBUG

```

```

#define new DEBUG_NEW

```

```

#undef THIS_FILE

```

```

static char THIS_FILE[] = __FILE__;
#ifdef

////////////////////////////////////
// CAboutDlg dialog used for App About
CWordCategorizationDlg(CWnd* pParent /*=NULL*/)
: CDialog(CAboutDlg::IDD, pParent)
{
public:
    CAboutDlg();

// Dialog Data
//{{AFX_DATA(CAboutDlg)
enum { IDD = IDD_ABOUTBOX };
//}}AFX_DATA

// ClassWizard generated virtual function overrides
//{{AFX_VIRTUAL(CAboutDlg)
protected:
virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
//}}AFX_VIRTUAL

// Implementation
protected:
//{{AFX_MSG(CAboutDlg)
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)
{
    //{{AFX_DATA_INIT(CAboutDlg)
    //}}AFX_DATA_INIT
}

void CAboutDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    //{{AFX_DATA_MAP(CAboutDlg)
    //}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CAboutDlg, CDialog)
    //{{AFX_MSG_MAP(CAboutDlg)
    // No message handlers
    //}}AFX_MSG_MAP

```



```
END_MESSAGE_MAP()
```

```
////////////////////////////////////
// CWordCategorizationDlg dialog
```

```
CWordCategorizationDlg::CWordCategorizationDlg(CWnd* pParent /*=NULL*/)
: CDialog(CWordCategorizationDlg::IDD, pParent)
```

```
{
   //{{AFX_DATA_INIT(CWordCategorizationDlg)
    m_csWords = _T("");
    m_csCategory = _T("");
    m_csWords2 = _T("");
   //}}AFX_DATA_INIT
    // Note that LoadIcon does not require a subsequent DestroyIcon in Win32
    m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}
```

```
void CWordCategorizationDlg::DoDataExchange(CDataExchange* pDX)
```

```
{
    CDialog::DoDataExchange(pDX);
   //{{AFX_DATA_MAP(CWordCategorizationDlg)
    DDX_Control(pDX, IDC_CATEGORY, m_ctCategory);
    DDX_Text(pDX, IDC_WORDS, m_csWords);
    DDX_Text(pDX, IDC_CATEGORY, m_csCategory);
    DDX_Text(pDX, IDC_WORDS2, m_csWords2);
   //}}AFX_DATA_MAP
}
```

```
BEGIN_MESSAGE_MAP(CWordCategorizationDlg, CDialog)
```

```
   //{{AFX_MSG_MAP(CWordCategorizationDlg)
    ON_WM_SYSCOMMAND()
    ON_WM_PAINT()
    ON_WM_QUERYDRAGICON()
    ON_BN_CLICKED(IDEXIT, OnExit)
    ON_BN_CLICKED(IDC_CLEAR, OnClear)
    ON_BN_CLICKED(IDVERIFY, OnVerify)
    // ON_BN_CLICKED(IDC_STEM, OnStem)
    ON_BN_CLICKED(IDC_DATAENTRY, OnDataentry)
   //}}AFX_MSG_MAP
END_MESSAGE_MAP()
```

```
////////////////////////////////////
// CWordCategorizationDlg message handlers
```

```
BOOL CWordCategorizationDlg::OnInitDialog()
```

```
{
    // To draw the icon, for MFC applications using the document/view model,

```

```

CDialog::OnInitDialog();

// Add "About..." menu item to system menu.

// IDM_ABOUTBOX must be in the system command range.
ASSERT((IDM_ABOUTBOX & 0xFFF0) == IDM_ABOUTBOX);
ASSERT(IDM_ABOUTBOX < 0xF000);

CMenu* pSysMenu = GetSystemMenu(FALSE);
if (pSysMenu != NULL)
{
    CString strAboutMenu;
    strAboutMenu.LoadString(IDS_ABOUTBOX);
    if (!strAboutMenu.IsEmpty())
    {
        pSysMenu->AppendMenu(MF_SEPARATOR);
        pSysMenu->AppendMenu(MF_STRING, IDM_ABOUTBOX,
strAboutMenu);
    }
}

// Set the icon for this dialog. The framework does this automatically
// when the application's main window is not a dialog
SetIcon(m_hIcon, TRUE);           // Set big icon
SetIcon(m_hIcon, FALSE);          // Set small icon

// TODO: Add extra initialization here

return TRUE; // return TRUE unless you set the focus to a control
}

void CWordCategorizationDlg::OnSysCommand(UINT nID, LPARAM lParam)
{
    if ((nID & 0xFFF0) == IDM_ABOUTBOX)
    {
        CAboutDlg dlgAbout;
        dlgAbout.DoModal();
    }
    else
    {
        CDialog::OnSysCommand(nID, lParam);
    }
}

// TODO: to clear all the string in the edit boxes

// If you add a minimize button to your dialog, you will need the code below
// to draw the icon. For MFC applications using the document/view model,

```


// this is automatically done for you by the framework.

```
void CWordCategorizationDlg::OnPaint()
```

```
{
    if (IsIconic())
    {
        CPaintDC dc(this); // device context for painting

        SendMessage(WM_ICONERASEBKGND,           (WPARAM)
dc.GetSafeHdc(), 0);

        // Center icon in client rectangle
        int cxIcon = GetSystemMetrics(SM_CXICON);
        int cyIcon = GetSystemMetrics(SM_CYICON);
        CRect rect;
        GetClientRect(&rect);
        int x = (rect.Width() - cxIcon + 1) / 2;
        int y = (rect.Height() - cyIcon + 1) / 2;

        // Draw the icon
        dc.DrawIcon(x, y, m_hIcon);
    }
    else
    {
        CDialog::OnPaint();
    }
}
```

// The system calls this to obtain the cursor to display while the user drags
// the minimized window.

```
HCURSOR CWordCategorizationDlg::OnQueryDragIcon()
```

```
{
    return (HCURSOR) m_hIcon;
}
```

```
void CWordCategorizationDlg::OnExit()
```

```
{
    CDialog::EndDialog(1);
}
```

```
void CWordCategorizationDlg::OnClear()
```

```
{
    // TODO: to clear all the string in the edit boxes
    m_csWords.Empty();
    m_csWords2.Empty();
}
```

```

m_csCategory.Empty();
GetDlgItem(IDC_STATIC_STATUS)->SetWindowText("Processing.....");
UpdateData(FALSE);

```

```

}
static CFont m_cFont;
void CWordCategorizationDlg::OnVerify()
{

```

```

    // TODO: to categorize text into specify category
    // int nStore;
    m_cFont.CreateFont(25,7,0,0,FW_BOLD,0,0,0,DEFAULT_CHARSET,0,0,0,0,
    _T("Arial")); //to set category box font

```

```

    int nMinLength = 10; // The Minimum no of characters to accept
    CString csWords4; //temp variable to store Processed data
    UpdateData(TRUE); // To Update the data from the control to member
    variable

```

```

    m_csWords2 = m_csWords; // Storing the input text to Processed index
    variable( m_csWords2)

```

```

    int nLength = m_csWords2.GetLength(); // Get the lenght of the input text &
    store it in int variable nLength

```

```

    if ( nLength >= nMinLength ) // if the nLength is Greater than Minimum
    Lenght say Lenght Verified Success
    {

```

```

        AfxMessageBox("Length Verified Success ");
        GetDlgItem(IDC_STATIC_STATUS)->SetWindowText("Length
        Verified Success.");

```

```

        m_csWords2.MakeLower(); // Convert Upper case to Lower Case call
        CString Member Function MakeLower()

```

```

        UpdateData(FALSE); // Update the data from the Variable to the
        Edit Control

```

```

        CString csTemp;

```

```

        int nSpos,nCnt;

```

```

        nSpos = 0 ;

```

```

        nCnt = 0; // a Character Array to store the special character constants that

```

you want to delete

```

        char cChar[70] =

```

```

        {' ',' ','0','1','2','3','4','5','6','7','8','9',' ','@','#','$','%','^','&','*','(',')','_','+','=','-'
        ',';','/','\','|',',',';','/','[','{','}','\"','/','?','!','>','<',' ','~'};

```

```

        int nNo = 0;

```

```

        int n = 0;

```

```

        AfxMessageBox(m_csWords2);

```



```

GetDlgItem(IDC_STATIC_STATUS)->SetWindowText("deleting
special chars");
while(n <10) // This while loop is to find & delete all the special
characters in the input text
{
    while ( nNo < 42 )
    {
        while(nSpos <= m_csWords2.GetLength())
        {
            nCnt = m_csWords2.Find(cChar[nNo],nSpos);
            if (nCnt != -1)
            {
                m_csWords2.Delete(nCnt,1);

                nSpos++ ;
            }
            nNo++;
            nSpos = 0;
        }
        n++;
        nNo = 0;
    }
    UpdateData(FALSE);
    nSpos = 0;

    AfxMessageBox("finished deleting special chars");

////////////////////////////////////
    CString m_csTemp;
    m_csTemp = m_csWords2;
    //    CString csTemp;    // local CString variable
    csTemp.Empty();
    CString csTemp2;
    int nstore = 0; // local int declaration
    int nTemp;    // local int declaration

    int nWords = 0;// total no of words
    unsigned int nTot = 0 ; // total of score
    FILE *fStream;
    //open txt file and read
    AfxMessageBox("Opening File Keyword.txt ");
    if((fStream = fopen("C:\\Project\\Keyword.txt","r")) == NULL).
    {
        AfxMessageBox("Cannot Open file Keyword.txt or file not
found");
        return;
    }

```

```

    }
    else
    {

```

```
        AfxMessageBox("KeyWord.txt open success");
```

```
        //to compare keyword in edit box(IDC_WORDS) with
```

```
keyword.txt
```

```
        for( nCnt=0; nCnt <= m_csTemp.GetLength(); nCnt++ )
        {

```

```
            CString csSorce,csSplit; // local CString variable
```

```
            nTemp = m_csTemp.Find(' ',nCnt );
```

```
            if(nTemp == -1 )
```

```
            {
```

```
                csTemp = m_csTemp;
```

```
            }
```

```
            else
```

```
            {
```

```
                csTemp = m_csTemp.Mid( 0, nTemp ); //
```

```
m_csTemp member variable for edit trol
```

```
                int nlen = m_csTemp.GetLength() -
```

```
csTemp.GetLength() ;// nlen int variable
```

```
                csSplit
```

```
m_csTemp.Right(m_csTemp.GetLength() - nTemp); // csSplit localvariable
```

```
                m_csTemp = csSplit;
```

```
                csTemp.TrimRight(); csTemp.TrimLeft(); //
```

```
Trimming for csTemp.
```

```
            }
```

```
            nstore += nTemp;
```

```
        fpos_t sPos = 0; // start of file
```

```
        fsetpos( fStream, &sPos );
```

```
        while(!feof(fStream) )
```

```
        {
```

```
            char szBuff[1500];
```

```
            fgets(szBuff,1500,fStream);
```

```
            CString csBuf,csSplit;
```

```
            csBuf.Format("%s",szBuff);
```

```
            CString csSt;
```



```

        csSt.Format("%s",szBuff);
        int nPos = csBuf.Find(',');
        csBuf = csBuf.Mid(0,nPos);
        csBuf.Remove(',');
        csTemp.Remove(' ');
        if( csBuf == csTemp )
        {
            csSplit= csSt.Right(5);
            // AfxMessageBox(csSplit);
            csSplit.Remove(' ');
            csSplit.Remove('0');csSplit.Remove('.');
            nTot += atol(csSplit);
            nWords++;
            csTemp2.Format("%i",nTot);
            AfxMessageBox(csTemp2);
            csWords4 += " " + csTemp;
            break;
        }
    }

    if(nTemp == -1)
        break;
}
//to calculate the category score
csTemp2.Format("nWords : %d, nTot: %d ",nWords,nTot);
AfxMessageBox(csTemp2);
int nVal=0;
if(nWords != 0)
{
    nVal = nTot / nWords;
}

if(nVal != 0)
{
    csTemp2.Format("0.%d",nVal);
    AfxMessageBox(csTemp2);
}

////////////////////////////////////

UpdateData(TRUE);
FILE *fStream1;

```

```

char szBuff[1500];

//open txt file and read
if((fStream1 = fopen("C:\\Project\\Category.txt","r")) == NULL)
{
    AfxMessageBox("Cannot Open file Category.txt");
    return;
}

CString csFind;
bool bFound = FALSE;
while(!feof(fStream1) )
{
    CString csStore,csTemp1,csT;
    fgets( szBuff,1500,fStream1 );
    csStore.Format("%s",szBuff);
    csFind = csStore.Right(6);
    csTemp1 = csFind.Left(3);
    CString csTi;
    csTi = csTemp2.Left(3);

    if( csTemp1 == csTi )
    {
        int nTemp1 = csStore.Find(',',0);
        csFind = csStore.Mid(0,nTemp1);
        bFound = TRUE;
        break;
    }
}

if(!bFound)
{ csFind = "General- Non Specific Category";
}

fclose(fStream1);
GetDlgItem(IDC_CATEGORY)->SetFont(&m_cFont);
m_csCategory = csFind;

}
else
{
    GetDlgItem(IDC_CATEGORY)->SetFont(&m_cFont);
    m_csCategory = "No Specific Category ";
    UpdateData(FALSE);
}

```



```

5.4.3.2.3 GetDlgItem(IDC_STATIC_STATUS)->SetWindowText("Processing
Success");
           m_csWords2 = csWords4;
           UpdateData(FALSE);
       }
   }
   else
   {
       GetDlgItem(IDC_STATIC_STATUS)->SetWindowText("Less Words
to Process");
       AfxMessageBox("Please Input more sentences: Error Less Words for
Processing");
   }
}

```

```

#include "stdafx.h"
#include "WordCategorization.h"
#include "KeyWordDlg.h"

void CWordCategorizationDlg::OnDataentry()
{
    // TODO: to call keyword data entry dialog
    CKeywordDlg dlg;

    int nResponse = dlg.DoModal();
    if (nResponse == IDOK)
    {
    }
    else if (nResponse == IDCANCEL)
    {
    }
}

```

5.4.3.2.2

Dokemntasi Aturcara KeywordDlg.cpp

```

/*****
*****

```

Copyright (c) Hazyzyizuan Abu Hassan

Project : Text Categorization

FileName : KeyWordDlg.cpp

Comments : Implementation fail -To get input from the user and insert data into database Keyword.txt

Written by : Hazyzyizuan Abu Hassan
hazyzy77@yahoo.com

```

*****

```

```

*****/

```

```

#include "stdafx.h"
#include "WordCategorization.h"
#include "KeyWordDlg.h"

```

```

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

```

```

////////////////////////////////////
// CKeyWordDlg dialog

```

```

CKeyWordDlg::CKeyWordDlg(CWnd* pParent /*=NULL*/)
: CDialog(CKeyWordDlg::IDD, pParent)

```

```

{
   //{{AFX_DATA_INIT(CKeyWordDlg)
    m_csKeyWord = _T("");
    m_csScore = _T("");
    m_csCategory = _T("");
    //}}AFX_DATA_INIT
}

```

```

void CKeyWordDlg::DoDataExchange(CDataExchange* pDX)

```



```

{
    CDialog::DoDataExchange(pDX);
   //{{AFX_DATA_MAP(CKeyWordDlg)
    DDX_Text(pDX, IDC_KEYWORD, m_csKeyWord);
    DDX_Text(pDX, IDC_SCORE, m_csScore);
    DDX_Text(pDX, IDC_CATEGORY, m_csCategory);
   //}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CKeyWordDlg, CDialog)
   //{{AFX_MSG_MAP(CKeyWordDlg)
    ON_BN_CLICKED(IDADD, OnAdd)
    ON_BN_CLICKED(IDQUIT, OnQuit)
   //}}AFX_MSG_MAP
END_MESSAGE_MAP()

////////////////////////////////////
// CKeyWordDlg message handlers

void CKeyWordDlg::OnAdd()
{
    // TODO: Add your control notification handler code here
    UpdateData(TRUE);
    CString csTemp; //assign csTemp as CString
    FILE *fStream2;
    if ((fStream2 = fopen("C:\\Project\\Keyword.txt", "a+w")) == NULL) //open txt
file, read and write to it.
    {
        AfxMessageBox("Cannot Open file Keyword.txt");
        return;
    }
    else
    {
        csTemp.Format("%s,%s,%s", m_csKeyWord, m_csCategory, m_csScore); //add
keyword,sport.score in database format
        csTemp = "\n" + csTemp;

        fputs( csTemp, fStream2);

        //AfxMessageBox("fopen success");
    }
    fclose( fStream2);
    m_csKeyWord.Empty(); //to clear string the edit boxes

```

```
Score.Empty();
ateData(FALSE);

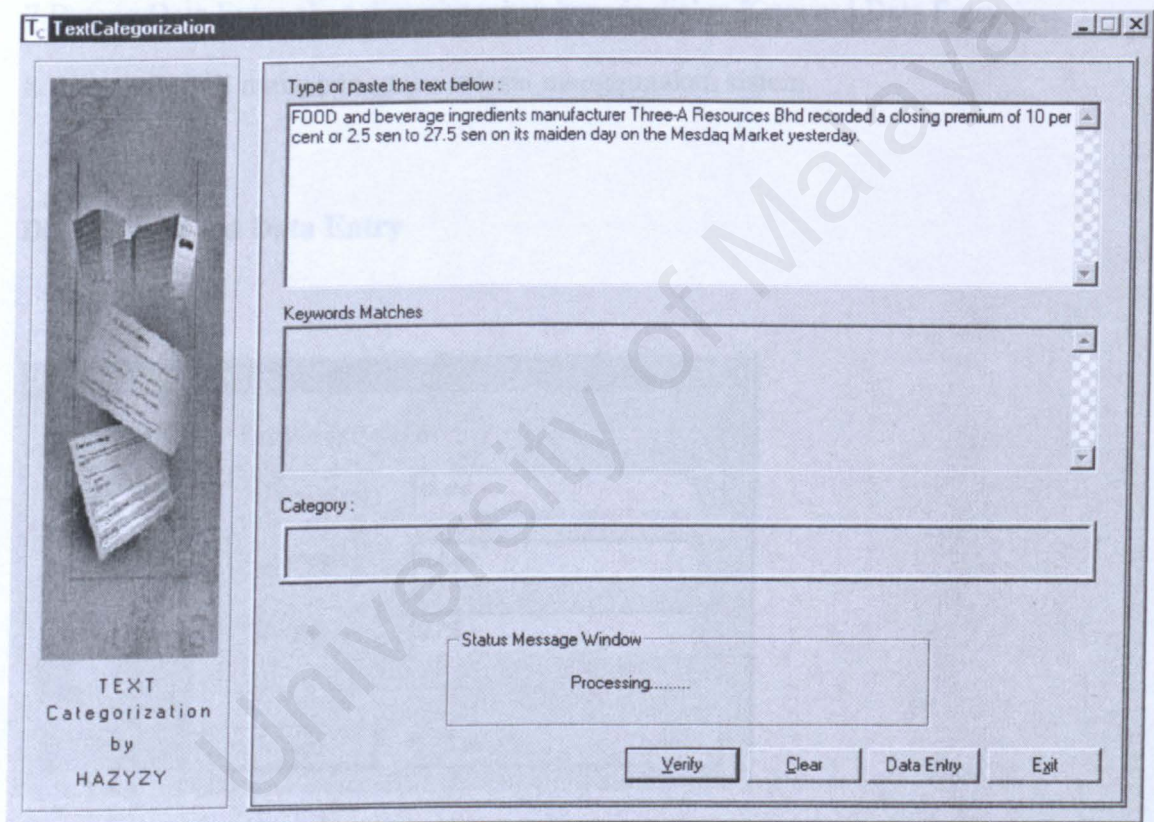
WordDlg::OnQuit()

log::OnCancel();
```


5.4.3.4 Dokumentasi Pengguna

Dokumentasi Pengguna pula akan menerangkan bagaimana hendak menggunakan sistem serta fungsi setiap butang.

Dialog Text Categorization



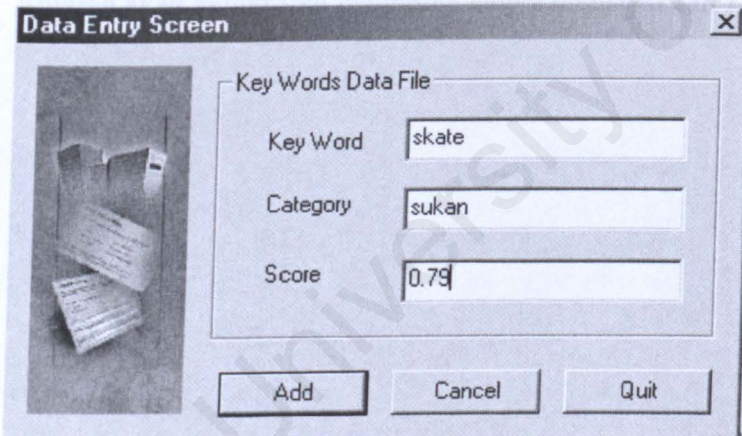
Rajah 5.11 : Dialog Text Categorization

1. Teks akan dimasukkan ke dalam kotak edit Type or Paste Below samada secara menaip atau secara salin dan letak.
2. Pada kotak edit Keyword Matches akan menunjukkan sekiranya ada keyword

pada teks yang dimasukkan yang sama dengan keyword di dalam pengkalan data.

3. Kotak edit Category akan menunjukkan kumpulan kategori utama.
4. Kotak teks Status Message Window akan menunjukkan proses yang berjalan.
5. Butang Verify akan memulakan proses pengkategorian .
6. Butang Clear akan mengosongkan semua teks di dalam kotak edit Type or Paste Below ,Keywords Matches dan Category.
7. Butang Data Entry akan disambungkan kepada dialog Keyword Data Entry.
8. Butang Exit di mana pengguna selesai menggunakan sistem.

Dialog Keyword Data Entry



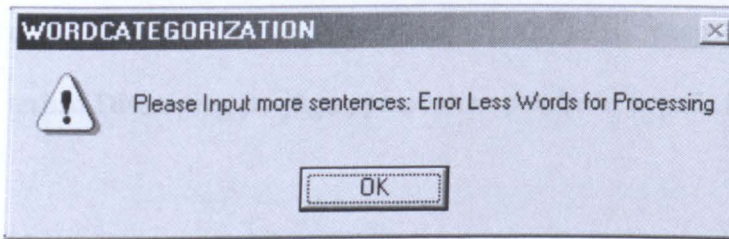
Rajah 5.12 : Dialog Keyword Data Entry

Dialog ini direka bagi memudahkan pengguna menambahkan keyword ke dalam pengkalan data.

1. Pengguna akan memasukkan keyword ke dalam kotak edit keyword.
2. Masukkan jenis kategori ke dalam kotak edit bagi category.
3. Masukkan pemberat ke dalam kotak edit bagi score.

4. Butang Add akan memasukkan ketiga-tiga data ke dalam pengkalan data.
5. Butang Cancel untuk memadamkan atau mengosongkan semua data yang tertulis dalam kotak edit.
6. Butang Quit untuk keluar dari dialog Data Entry

Dialog Amaran



Rajah 5.13 : Dialog Amaran

Amaran akan keluar jika pengguna hanya memasukkan 1 perkataan sahaja.

5.5 Kelemahan Dan Kebaikan

Terdapat beberapa kebaikan yang dikenalpasti setelah pembangunan sistem ini . Diantaranya ialah masa yang diambil untuk mengkategorikan teks adalah pendek.

Walaupun sistem ini telah dapat dibangunkan namun terdapat beberapa kelemahan yang dikenalpasti. Diantaranya ialah pembuangan kata imbuhan pada kata dasar setiap perkataan. Jumlah perkataan yang boleh dimasukkan adalah terhad. Disamping itu juga , pemberian pemberat kepada keyword.

1. Memastikan struktur program untuk while/for dan if disusai bagi merendahkan perisian untuk membaca aturcara.
2. Menambahkan fungsi pada aturcara.
3. Menentukan nilai pemberat bagi keyword

Penyelenggaraan yang berkesan amatlah penting bagi memastikan segala perubahan yang dibuat diuruskan dengan betul. Penyelenggaraan yang berkesan boleh menjamin kualiti sesebuah sistem. Oleh itu amatlah penting bagi kita untuk mengikut keteksidan penyelenggaraan bagi melihat sejauh mana kejayaan aktiviti tersebut.

BAB 6 : PENYELENGGARAAN

Pembangunan sistem adalah selesai bila sistem berfungsi sepenuhnya, iaitu apabila pengguna menggunakan sistem pada persekitaran yang sebenar. Apa-apa kerja melakukan perubahan yang dilakukan selepas operasi adalah dianggap *menyelenggara*. Penyelenggaraan merupakan satu proses di mana aktiviti-aktiviti perubahan yang dilakukan ke atas sistem ditangani dengan memperkenalkan penyelenggaraan yang sesuai. Perubahan-perubahan tersebut perlu dilakukan disebabkan oleh beberapa factor seperti pembetulan, penyesuaian, penyempurnaan dan pencegahan.

Penyelenggaraan pembetulan merujuk kepada perubahan yang dibuat ke atas sistem disebabkan oleh kesilapan atau pepijat. Walaupun ianya telah diuji di fasa pembangunan dan implementasi namun, perkara ini sering berlaku dan dianggap biasa. Antara penyelenggaraan yang dibuat selepas pembangunan sistem ialah :

1. Memastikan struktur gelungan untuk *while*, *for* dan *if* disusun bagi memudahkan pembangun untuk membaca aturcara.
2. Menambahkan fungsi pada aturcara.
3. Menukarkan nilai pemberat bagi keyword

Penyelenggaraan yang berkesan amatlah penting bagi memastikan segala perubahan yang dibuat diuruskan dengan betul. Penyelenggaraan yang berkesan boleh menjamin kualiti sesebuah sistem. Oleh itu amatlah penting bagi kita untuk mengukur keberkesanan penyelenggaraan bagi melihat sejauh mana kejayaan aktiviti tersebut.

BAB 7 : PEMBANGUNAN MASA HADAPAN

Pengenalan dan definisi keseluruhan kepada projek diterang secara umum bagi mendapat gambaran awal projek ini. Didalam laporan ini juga membincangkan kajian yang telah dilakukan semasa fasa analisis dan rekabentuk projek ini. Kajian yang dilakukan adalah seperti penggunaan teknik untuk pengkategorian, perisian dan idea yang diperlukan untuk membangunkan Sistem Pengkategorian Teks. Teknik yang dipilih adalah Nearest Neighbour dan juga pemprosesan indeks digunakan untuk mengekstrakkan maklumat daripada artikel.

Sistem yang dibangunkan merupakan sistem yang pertama dibangunkan oleh pembangun dan ianya masih memerlukan pelbagai fungsi bagi menjamin kesempurnaan sistem. Sistem ini dibangunkan berdasarkan objektif dan skop yang digariskan dalam laporan awal. Kajian sedia ada, dan maklumat-maklumat yang lain yang diperolehi telah dijadikan panduan dalam menyiapkan projek ini. Metodologi yang dipilih banyak membantu dalam membuat perancangan projek.

Setelah pembangunan sistem selesai masih terdapat lagi beberapa kekurangan pada fungsi yang telah dibina. Namun kekurangan ini perlu dibaiki lagi demi menyempurnakan lagi sistem ini. Diantara kekurangan yang dikenalpasti ialah pembuangan kata imbuhan pada kata dasar setiap perkataan. Disamping itu juga, pemberian pemberat kepada keyword perlu dikaji supaya pengkategorian menjadi lebih sempurna. Selain daripada itu had jumlah perkataan yang dimasukkan juga perlu dikaji.

Penggunaan pelbagai bahasa di dalam sistem ini perlu ditekankan agar sistem ini dapat digunakan secara meluas .Akhir sekali diharap pembangunan sistem ini akan dapat memenuhi kehendak pengguna.

1. P. Sellappan, *Software Engineering Management And Method 1st Edition* , Sejana Publishing 2000.
2. Kenuli, K E & Kenuli, L.B, *System Analysis and Design 4th Edition* , Prentice Hall Inc, Upper Saddle River, NJ, 1999.
3. Pflieger, S L, *Software Engineering Theory And Practice 2th Edition* , Prentice Hall Inc, Upper Saddle River, NJ, 2001.
4. George F.Luger,William A.Stubblefield, *Artificial Intelligence : Structures And Strategies For Complex Problem Solving 3rd Edition* , Addison Wesley Longman Inc,1996.
5. Peter Jackson, *Introduction to Expert Systems 3rd Edition* , Addison Wesley Longman Inc,1999.
6. P.Sellappan, *Visual C++ Through Example*, First Published 1999.
7. Mohamad Noorhan, Merrik, Safaw, Abdul Rahman, Kamarulrifia Abdul Jalil , *Analisis Dan Rekabentuk Sistem Makhawat* , McGraw-Hill (M) Sdn Bhd. , 2001.
8. Allen, J, *Natural Language Understanding 3rd Edition* The Benjamin/ Cummings Publishing Company Inc,1993.
9. P.C.James, W.D.Jack, *C++ Program Design 2nd Edition* , McGraw-Hill Companies 1999.

BIBLIOGRAFI

1. P. Sellapan, Software Engineering Management And Method 1st Edition , Sejana Publishing 2000.
2. Kendall, K.E & Kendall, J.E, System Analysis and Design 4th Edition , Prentice Hall Inc, Upper Saddle River, NJ, 1999.
3. Pfleeger ,S.L,Software Engineering Teory And Practice ,2nd Edition , Prentice Hall Inc, Upper Saddle River, NJ, 2001.
4. George F.Luger,William A.Stubblefield .Artificial Intelligence : Structures And Strategies For Complex Problem Solving, 3rd Edition , Addison Wesley Longman,Inc,1998.
5. Peter Jackson,Introduction to Expert Systems, 3rd Edition , Addison Wesley Longman,Inc,1999.
6. P.Sellapan,Visual C++-Through Example,First Published 1999.
7. Mohamad Noorman Marrek,Safawi Abdul Rahman,Kamarulariffin Abdul Jalil , Analisis Dan Rekabentuk Sistem Maklumat ,McGraw-Hill (M) Sdn.Bhd, , 2001.
8. Allen,J,Natural Language Understanding,2nd Edition,The Benjamin/ Cummings Publishing Company,Inc,1995.
9. P.C.James,W.D.Jack,C++ Program Design, 2nd Edition, McGraw-Hill Companies,1999.

10. Deitel & Deitel, C++ How To Program, 2nd Edition, Prentice Hall, 1998
11. Moulinier, I., A Framework for Comparing Text Categorization Approaches, 1997
12. David D. Lewis, A Comparison of 2 Learning Algorithms for Text Categorization, April, 1994.
13. Terry R. Payne, Dimensionality Reduction And Representation For Nearest Neighbour Learning, 1999.
14. Ufuk Ilhan, Application Of K-NN And FPTC Based Text Categorization Algorithms To Turkish News Report. 2001.
15. Vipin Kumar, Karypis, G., Eui Hong Han, Text Categorization Using Weight Adjusted K-Nearest Neighbor Classification, March 20, 1999.
16. David M. Mount, ANN Programming Manual, 1998.
17. William B. Cavnar, John M. Trenkle, N-Gram-Based Text Categorization. 1999
18. <http://www.nstp.com.my>
19. <http://www.programmersheaven.com>
20. <http://www.cplusplus.com>
21. <http://www.nstp.com.my>

22. <http://msdn.microsoft.com/visualc/>
23. <http://www.codeguru.com>
24. <http://www.aul.fiu.edu/tech/visualc.html>
25. <http://www.devcentral.iftech.com/articles/mfc/default.php>
26. <http://www.fucntionx.com/visualc/>
27. <http://www.gotdotnet.com/team/cplusplus/>
28. <http://www.1001tutorials.com/vcpp/index>